

# BloodNet

## Problem Statement:

### BloodNet-CRM-for-Blood-and-Plasma-Management

Access to blood and plasma during emergencies is often delayed due to fragmented donor records, manual request handling, and lack of real-time visibility into inventory. Hospitals rely on outdated processes such as phone calls, paperwork, and disconnected databases, leading to critical time loss in life-saving situations. Additionally, donors are not effectively engaged, resulting in missed opportunities for timely donations and poor inventory management in blood banks..

## Overview

The proposed **BloodNet CRM** is a Salesforce-based system that centralizes and automates the entire blood and plasma management process, the solution provides:

- A **central donor registry** with blood group, eligibility status, and donation history.
- A **hospital request management system** to raise and track urgent needs.
- **Automated donor matching** to identify compatible donors instantly.
- **Email/SMS alerts** to donors and hospitals for faster communication.
- **Real-time inventory dashboards** to monitor blood stock and trends.
- **Reports and analytics** for healthcare administrators to make data-driven decisions.

This solution eliminates manual inefficiencies, accelerates emergency response, and improves donor engagement, ultimately saving lives and strengthening healthcare delivery.

# PHASE 1: Problem Understanding & Industry Analysis

## **Requirement Gathering**

The following key requirements were identified through analysis of existing challenges in blood and plasma management:

1. **Centralized Donor Records** – Store and maintain donor information, including blood group, eligibility, and donation history.
2. **Hospital Request System** – Enable hospitals and clinics to raise urgent requests for blood and plasma.
3. **Automated Donor Matching** – Match requests with compatible donors based on blood group and location.
4. **Communication Mechanism** – Provide automated alerts and notifications (SMS/Email) to donors and hospitals.
5. **Inventory Management** – Track available blood stock, plasma units, and expiry dates across multiple blood banks.
6. **Reporting & Analytics** – Generate dashboards and reports for monitoring trends, donor engagement, and fulfillment rates.

## **Stakeholder Analysis**

The solution involves multiple stakeholders, each with unique needs:

- **Donors** – Want a simple way to register, update their eligibility, and receive reminders/alerts.

- **Hospitals & Clinics** – Require a fast and reliable method to raise requests and track fulfillment.
- **Blood Banks** – Need to manage available stock, expiry dates, and donations efficiently.
- **Healthcare Administrators/Government Bodies** – Require reports and dashboards for decision-making and monitoring regional health preparedness.

## **Business Process Mapping**

The current manual approach involves phone calls, fragmented spreadsheets, and delayed coordination, which leads to inefficiency. The proposed CRM streamlines the workflow:

1. **Donor Registration** → Donor enters details such as blood group, contact, and last donation date.
2. **Hospital Request** → Hospital submits a requirement specifying blood/plasma type and urgency.
3. **Automated Matching** → CRM identifies compatible donors and notifies them.
4. **Inventory Update** → Stock levels are updated when requests are fulfilled.
5. **Reporting** → Dashboards display live statistics on requests, fulfillment, and inventory.

## **Industry-Specific Use Case Analysis**

- **Healthcare Context:** Emergencies such as road accidents, surgeries, or pandemics demand quick access to compatible blood and plasma.
- **Challenges in Industry:** Lack of integration between donor databases, inefficient communication, and outdated tracking methods.
- **CRM Advantage:** Salesforce CRM provides automation, real-time dashboards, and donor lifecycle management, which are missing in traditional approaches. This ensures faster emergency response and improved healthcare efficiency.

## AppExchange Exploration

- **Existing Apps:** Salesforce AppExchange offers generic healthcare and donor management applications (e.g., Health Cloud, fundraising solutions).
- **Gap Identified:** Most available apps are broad, complex, or costly, making them unsuitable for smaller hospitals and NGOs.
- **Proposed Differentiator:** **BloodNet CRM** is designed as a targeted, affordable, and scalable solution focusing exclusively on **blood and plasma management**.

## PHASE 2: Org Setup & Configuration

### 1. Salesforce Editions

We are using a Salesforce Developer Edition as the build and test environment for BloodNet CRM. This org provides full access to custom objects, Flows, Lightning App Builder, Reports & Dashboards required for the project.

### 2. Company Profile Setup

Setup → *Company Settings* → **Company Information** → click **Edit**.

- Company Name / Organization Name: BloodNet CRM
- Default Time Zone: GMT+05:30 Asia/Kolkata
- Default Currency: USD
- Locale / Language: English (India)

The screenshot shows the Salesforce Setup interface. On the left, a sidebar titled 'Company Settings' has 'Business Hours' selected. The main content area is titled 'Company Information' and shows the profile for 'BloodNet CRM'. It includes sections for 'Organization Detail' (with fields like Organization Name, Primary Contact, Division, Address, Fiscal Year Starts In), 'User Licenses' (listing various user types and their status), and 'System Metrics' (like API Requests, Streaming API Events, Restricted Logins). At the bottom, it shows 'Created By' and 'Modified By' information.

### 3. Business Hours & Holidays

#### What to create / do

- Define the operating hours and add important holidays.

#### How (where to click)

- Setup → *Company Settings* → **Business Hours** → New
  - Name: BloodNet Standard Hours
  - Time Zone: GMT+05:30 Asia/Kolkata
  - Working Hours: Mon–Sat 09:00 AM – 09:00 PM
  - Save.

**Organization Business Hours**

Select the days and hours that your support team is available. These hours, when associated with escalation rules, determine the times at which cases can escalate. If you enter blank business hours for a day, that means your organization does not operate on that day.

**Business Hours Edit**

**Step 1. Business Hours Name**

Business Hours Name: BloodNet Standard Hours  
Active:

**Step 2. Time Zone**

Time Zone: (GMT+05:30) India Standard Time (Asia/Kolkata)

**Step 3. Business Hours**

Day	From	To	24 hours
Sunday	12:00 AM	12:00 AM	<input type="checkbox"/>
Monday	9:00 AM	9:00 PM	<input type="checkbox"/>
Tuesday	9:00 AM	9:00 PM	<input type="checkbox"/>
Wednesday	9:00 AM	9:00 PM	<input type="checkbox"/>
Thursday	9:00 AM	9:00 PM	<input type="checkbox"/>
Friday	9:00 AM	9:00 PM	<input type="checkbox"/>
Saturday	9:00 AM	9:00 PM	<input type="checkbox"/>

Save Cancel

Setup → *Company Settings* → **Holidays** → **New Holiday** → add national holidays

**Holidays**

Holidays are dates and times at which business hours are suspended. Business hours are the days and hours that your support team is available.

**Holidays**

Action	Holiday Name	Description	Date and Time
Edit   Del	Christmas	An annual festival commemorating the birth of Jesus Christ	9/1/2026 All Day
Edit   Del	Independence Day	National Holiday – Offices Closed	8/15/2026 All Day

**Elapsed Holidays**

No records to display

**Organization Business Hours**

Select the days and hours that your support team is available. These hours, when associated with escalation rules, determine the times at which cases can escalate.

If you enter blank business hours for a day, that means your organization does not operate on that day.

Holiday Name	Description	Date and Time
Christmas	An annual festival commemorating the birth of Jesus Christ	9/1/2026 All Day
Independence Day	National Holiday – Offices Closed	8/15/2026 All Day

Wednesday 9:00 AM to 9:00 PM  
Thursday 9:00 AM to 9:00 PM  
Friday 9:00 AM to 9:00 PM  
Saturday 9:00 AM to 9:00 PM

Active ✓  
Created By Akshat.Kumar 9/23/2025, 5:35 AM Last Modified By Akshat.Kumar 9/23/2025, 5:54 AM

**Holidays**

Holiday Name	Description	Date and Time
Christmas	An annual festival commemorating the birth of Jesus Christ	9/1/2026 All Day
Independence Day	National Holiday – Offices Closed	8/15/2026 All Day

[Edit](#)

[Back To Top](#) Always show me **more** records per related list

## 4. Fiscal Year Settings

Setup → *Company Settings* → **Fiscal Year** → ensure **Standard Fiscal Year** is selected

Fiscal settings are set to align statistical reports and year-on-year donor analytics with regional reporting cycles.

**Organization Fiscal Year Edit: BloodNet CRM**

To specify the fiscal year type for your organization, choose one of the options below.

**Fiscal Year Information**  
Your organization can change the fiscal year start month, and specify whether the fiscal year name is set to the starting or ending year. For example, if your fiscal year starts in April 2025 and ends in March 2026, your Fiscal Year setting can be either 2025 or 2026.

Changing the fiscal year shifts fiscal periods and impacts opportunities and forecasts across your organization. If your forecast periods are set to quarterly, adjusting the fiscal year start month will erase existing forecast adjustments and quotas. Consider exporting a data backup before implementing this change.

**Change Fiscal Year Period**

Name	BloodNet CRM
Fiscal Year Start Month	January
Fiscal Year is Based On	<input checked="" type="radio"/> The starting month

[Save](#) [Cancel](#)

Didnt find what you're looking for? Try using Global Search.

## 4. User Setup & Licenses

### Profiles vs Roles vs Users (Quick Recap)

- Profile → Defines what a user can do (object access, field access, tabs).
- Role → Defines what records they can see (record visibility in the hierarchy).
- User → A person account in Salesforce, which is always assigned one Profile + optionally one Role.
- Created users:
  - System Administrator (Salesforce platform user)
  - Hospital Admin (Salesforce platform user)
  - Blood Bank Admin (Salesforce platform user)

The screenshot shows the Salesforce Setup interface for managing users. The left sidebar navigation includes links for Setup Home, Salesforce Go, Service Setup Assistant, Commerce Setup Assistant, Field Service Setup Home (Beta), Hyperforce Assistant, Release Updates, Salesforce Mobile App, Lightning Usage, Optimizer, Sales Cloud Everywhere, Administration, and a detailed section for Users. The main content area is titled "All Users" and displays a table of users with columns for Action, Full Name, Alias, Username, Role, Active, and Profile. The table lists several users, each with a unique profile and role assigned. The bottom of the page includes buttons for New User, Reset Password(s), and Add Multiple Users, along with a navigation bar for letters A through Z and an "All" link.

Action	Full Name	Alias	Username	Role	Active	Profile
<input type="checkbox"/>	Edit BloodNet_BloodBank_Profile	bank_adm	bankt@bloodnet.com	Blood Bank Admin	<input checked="" type="checkbox"/>	Standard Platform User
<input type="checkbox"/>	Edit BloodNet_Hospital_Profile	user	hospital1@bloodnet.com	Hospital User	<input type="checkbox"/>	Standard Platform User
<input type="checkbox"/>	Edit Chatter_Expert	Chatter	chatty@00dk000000bhz7uan.com	Chatter	<input checked="" type="checkbox"/>	Chatter Free User
<input type="checkbox"/>	Edit EPIC_OroFarm	OEPIIC	oeic.ba2016cc894@orofarm.salesforce.com	System Admin	<input checked="" type="checkbox"/>	System Administrator
<input type="checkbox"/>	Edit Hospital Admin	hosse_adm	akshat.kumar.aim22@jpsits.net	Hospital Admin	<input checked="" type="checkbox"/>	Standard Platform User
<input type="checkbox"/>	Edit Kumar.Akshat	aks	akshat.sheal304@openforce.com	System Admin	<input checked="" type="checkbox"/>	System Administrator
<input type="checkbox"/>	Edit User_Integration	integ	integration@00dk000000bhz7uan.com	System Admin	<input checked="" type="checkbox"/>	Analytics Cloud Integration User
<input type="checkbox"/>	Edit User_Security	sec	insightssecurity@00dk000000bhz7uan.com	System Admin	<input checked="" type="checkbox"/>	Analytics Cloud Security User

Setup → Users → Profiles → click Standard User (or another base) → Clone.

The screenshot shows the Salesforce Setup interface with the 'Profiles' page selected. The profile 'BloodNet\_BloodBank\_Profile' is displayed, which includes a description and creation details. The 'Page Layouts' section lists various global and object-specific page layouts, each with a 'View Assignment' link. The right side of the page shows location group assignments for different objects like Account, Opportunity, and Payment.

The screenshot shows the Salesforce Setup interface with the 'Profiles' page selected. The profile 'BloodNet\_Hospital\_Profile' is displayed, which includes a description and creation details. The 'Page Layouts' section lists various global and object-specific page layouts, each with a 'View Assignment' link. The right side of the page shows location group assignments for different objects like Account, Opportunity, and Payment.

## 6. Roles & Role Hierarchy

Setup → Users → Roles → Set Up Roles → Click Add Role under desired parent.

- Create: System Admin (top), Hospital Admin, Hospital User, BloodBank Admin, Donor Agent.

The screenshot shows the Salesforce Setup Roles page. The left sidebar is titled 'SETUP' and includes sections for 'Users' (Roles is selected) and 'PLATFORM TOOLS'. The main content area is titled 'Creating the Role Hierarchy' and displays a tree view of roles. The hierarchy is as follows:

- BloodNet CRM
  - System Administrator
  - Blood Bank Admin
    - Donor Agent
  - Hospital Admin
    - Hospital User

Each role node has 'Edit | Del | Assign' options. A 'Show in tree view' dropdown is visible at the top right of the hierarchy table.

## 7. Permission Sets

**Why:** Profiles handle most access, but Permission Sets let you add **extra rights without cloning profiles**. This makes your design look scalable and professional.

**What to create:**

- Permission Set 1 → Reporting Access → gives access to Dashboards/Reports.
- Permission Set 2 → Donor Communication → gives access to send emails/notifications.

**Permission Set**  
**Reporting Access**

**Permission Set Overview**

Description	License	API Name
Gives access to dashboards and reports.	Session Activation Required: <input type="checkbox"/>	Reporting_Access
	Permission Set Groups Added To: 0	Namespace Prefix: Akshat_Kumar
		Created By: Akshat Kumar, 9/23/2025, 2:15 PM
		Last Modified By: Akshat Kumar, 9/23/2025, 2:15 PM

**Apps**

**Assigned Apps**  
Settings that specify which apps are visible in the app menu

**Assigned Connected Apps**  
Settings that specify which connected apps are visible in the app menu

**Object Settings**  
Permissions to access objects and fields, and settings such as tab availability

**App Permissions**  
Permissions to perform app-specific actions, such as "Manage Call Centers"

**Apex Class Access**  
Permissions to execute Apex classes

**Visualforce Page Access**  
Permissions to execute Visualforce pages

**External Data Source Access**  
Permissions to authenticate against external data sources

**Flow Access**  
Permissions to execute Flows

**Named Credential Access**  
Permissions to authenticate against named credentials

**External Credential Principal Access**  
Permissions to authenticate with external credential principal mappings

**Service Presence Statuses Access**  
Permissions to access Service Presence Statuses

**Custom Permissions**  
Permissions to access custom processes and apps

**Custom Metadata Types**

**Permission Set**  
**Donor Communication**

**Permission Set Overview**

Description	License	API Name
Gives access to send emails/notifications.	Session Activation Required: <input type="checkbox"/>	Donor_Communication
	Permission Set Groups Added To: 0	Namespace Prefix: Akshat_Kumar
		Created By: Akshat Kumar, 9/23/2025, 2:16 PM
		Last Modified By: Akshat Kumar, 9/23/2025, 2:16 PM

**Apps**

**Assigned Apps**  
Settings that specify which apps are visible in the app menu

**Assigned Connected Apps**  
Settings that specify which connected apps are visible in the app menu

**Object Settings**  
Permissions to access objects and fields, and settings such as tab availability

**App Permissions**  
Permissions to perform app-specific actions, such as "Manage Call Centers"

**Apex Class Access**  
Permissions to execute Apex classes

**Visualforce Page Access**  
Permissions to execute Visualforce pages

**External Data Source Access**  
Permissions to authenticate against external data sources

**Flow Access**  
Permissions to execute Flows

**Named Credential Access**  
Permissions to authenticate against named credentials

**External Credential Principal Access**  
Permissions to authenticate with external credential principal mappings

**Service Presence Statuses Access**  
Permissions to access Service Presence Statuses

**Custom Permissions**  
Permissions to access custom processes and apps

**Custom Metadata Types**

Assigned to Hospital Admin and Blood Bank Admin test users.

The screenshot shows the Salesforce Setup interface. The left sidebar is titled 'ADMINISTRATION' and includes 'Users' which is expanded, showing 'Permission Sets'. The main content area is titled 'Reporting Access' and displays the 'Current Assignments' table. The table has columns for Full Name, Active, Role, Profile, User License, and Expires On. One row is present: Hospital Admin, Hospital Admin, Standard Platform User, Salesforce Platform.

Full Name	Active	Role	Profile	User License	Expires On
Hospital Admin	✓	Hospital Admin	Standard Platform User	Salesforce Platform	

The screenshot shows the Salesforce Setup interface. The left sidebar is titled 'ADMINISTRATION' and includes 'Users' which is expanded, showing 'Permission Sets'. The main content area is titled 'Donor Communication' and displays the 'Current Assignments' table. The table has columns for Full Name, Active, Role, Profile, User License, and Expires On. One row is present: BloodNet\_BloodBank\_Profile, Blood Bank Admin, Standard Platform User, Salesforce Platform.

Full Name	Active	Role	Profile	User License	Expires On
BloodNet_BloodBank_Profile	✓	Blood Bank Admin	Standard Platform User	Salesforce Platform	

## 8.Org-Wide Defaults (OWD)

OWD is **core to security**.

This will be configured in Phase 3, as the settings need to be applied to the custom objects that will be created then.

## 9.Login Access Policies

Login Access Policies determine who can log in as a user and for how long. This is useful for admin troubleshooting and security monitoring.

Setup → Quick Find → **Login Access Policies**.

The screenshot shows the Salesforce Setup interface. The left sidebar has 'Login' selected under 'Login Access Policies'. The main content area is titled 'Login Access Policies' and contains a sub-section 'Manage Support Options'. It shows a table with one row for 'Administrators Can Log in as Any User', which is checked and labeled 'Enabled'. Below this is another table for 'Support Organization' with a single row for 'Salesforce.com Support', which is checked and labeled 'Available to Users'. At the bottom are 'Save' and 'Cancel' buttons.

## PHASE 3: Data Modelling & Relationship

**Goal :** Design and implement a scalable data model in Salesforce to BloodNet Blood and Plasma management business processes.

### 1. Objects

- **Custom Objects Created:**

- Donor
- Hospital Staff

- Request
- Inventory

- **Standard Objects Used:**

- User (for Blood Bank Admin / System Admin)
- Account

## 2. Fields

- Donor → Blood Group, Contact, Email, Last Donation, Next Eligible Date, etc.
- Hospital Staff → Hospital Name, Hospital Staff Name, Staff Id, Contact, Email, Role Designation, etc.
- Request → Blood Group Needed, Request Date, Request Status, Units Required, Urgency, Fulfillment Date Lookups.
- Inventory → Blood Type, Units Available, Expiry, Storage Location, Managed By (Lookup to User).

Object Manager				
108+ Items, Sorted by Label				
Data Kit Deployment Log	DataKitDeploymentLog	Standard Object		
Data Use Legal Basis	DataUseLegalBasis	Standard Object		
Data Use Purpose	DataUsePurpose	Standard Object		
Delivery Estimation Setup	DeliveryEstimationSetup	Standard Object		
Digital Wallet	DigitalWallet	Standard Object		
Donor	Donor__c	Custom Object	Stores donor information (name, blood group, eligibility, contact).	9/24/2025
Duplicate Record Item	DuplicateRecordItem	Standard Object		
Duplicate Record Set	DuplicateRecordSet	Standard Object		
Email Message	EmailMessage	Standard Object		
Engagement Channel Type	EngagementChannelType	Standard Object		
Engagement Channel Work Type	EngagementChannelWorkType	Standard Object		
Entitlement	Entitlement	Standard Object		
Entitlement Contact	EntitlementContact	Standard Object		
Event	Event	Standard Object		
Finance Balance Snapshot	FinanceBalanceSnapshot	Standard Object		

Object Manager				
108+ Items, Sorted by Label				
Finance Transaction	FinanceTransaction	Standard Object		
Fulfillment Order	FulfillmentOrder	Standard Object		
Fulfillment Order Item Adjustment	FulfillmentOrderItemAdjustment	Standard Object		
Fulfillment Order Item Tax	FulfillmentOrderItemTax	Standard Object		
Fulfillment Order Product	FulfillmentOrderLineItem	Standard Object		
Guest User Anomaly Event Store	GuestUserAnomalyEventStore	Standard Object		
Hospital Satff	Hospital_Satff__c	Custom Object	Stores hospital users/staff who raise blood requests.	9/24/2025
Image	Image	Standard Object		
Incident	Incident	Standard Object		
Incident Related Item	IncidentRelatedItem	Standard Object		
Individual	Individual	Standard Object		
Inventory	Inventory__c	Custom Object	Stores blood stock available with Blood Bank Admin.	9/24/2025
Inventory Item Reservation	InventoryItemReservation	Standard Object		
Inventory Reservation	InventoryReservation	Standard Object		
Invoice	Invoice	Standard Object		
Invoice Line	InvoiceLine	Standard Object		
Lead	Lead	Standard Object		
Learning Item	LearningItem	Standard Object		
Legal Entity	LegalEntity	Standard Object		
List Email	ListEmail	Standard Object		

The screenshot shows the Salesforce Object Manager page. At the top, there are tabs for Setup, Home, and Object Manager. A search bar at the top right contains the text "Search Setup". Below the header, the title "Object Manager" is displayed, followed by "211 Items, Sorted by Label". On the right side of the main area, there are buttons for "Quick Find", "Schema Builder", and "Create". The main content area is a table listing various objects:

Recommendation	Recommendation	Standard Object		
Refund	Refund	Standard Object		
Refund Line Payment	RefundLinePayment	Standard Object		
Related Problem and Incident	ProblemIncident	Standard Object		
Report Anomaly Event Store	ReportAnomalyEventStore	Standard Object		
Request	Request_c	Custom Object	Stores hospital requests for blood/plasma, linked to both Staff & Donor.	9/24/2025
Resource Absence	ResourceAbsence	Standard Object		
Resource Preference	ResourcePreference	Standard Object		
Return Order	ReturnOrder	Standard Object		
Return Order Item Adjustment	ReturnOrderItemAdjustment	Standard Object		
Return Order Item Tax	ReturnOrderItemTax	Standard Object		
Return Order Line Item	ReturnOrderLineitem	Standard Object		
Scorecard	Scorecard	Standard Object		
Scorecard Association	ScorecardAssociation	Standard Object		
Scorecard Metric	ScorecardMetric	Standard Object		
Seller	Seller	Standard Object		
Service Appointment	ServiceAppointment	Standard Object		
Service Appointment Attendee	ServiceAppointmentAttendee	Standard Object		
Service Contract	ServiceContract	Standard Object		
Service Resource	ServiceResource	Standard Object		

## Fields and relationships in the created objects—

The screenshot shows the Salesforce Object Manager page for the "Donor" object. At the top, there are tabs for Setup, Home, and Object Manager. A search bar at the top right contains the text "Search Setup". Below the header, the title "SETUP > OBJECT MANAGER" and "Donor" are displayed, followed by "11 Items, Sorted by Field Label". On the right side of the main area, there are buttons for "Quick Find", "New", "Deleted Fields", "Field Dependencies", and "Set History Tracking". The main content area is a table titled "Fields & Relationships" with the following columns: FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. The table lists the fields for the Donor object:

	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Blood Group	Blood_Group__c	Picklist		
Lightning Record Pages	Contact Number	Contact_Number__c	Phone		
Buttons, Links, and Actions	Created By	CreatedById	Lookup(User)		
Compact Layouts	Donor Name	Name	Text(80)		
Field Sets	Eligibility Status	Eligibility_Status__c	Picklist		
Object Limits	Email	Email__c	Email		
Record Types	Last Donation Date	Last_Donation_Date__c	Date		
Related Lookup Filters	Last Modified By	LastModifiedById	Lookup(User)		
Restriction Rules	Next Eligible Date	Next_Eligible_Date__c	Formula (Date)		
Scoping Rules	Notes	Notes__c	Long Text Area(32768)		
Object Access	Owner	OwnerId	Lookup(User/Group)		
Triggers					
Flow Triggers					
Validation Rules					
Conditional Field Formatting					

Setup Home Object Manager

SEARCH Setup > OBJECT MANAGER Hospital Staff

**Fields & Relationships**  
10 items, Sorted by Field Label

	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Contact Number	Contact_Number__c	Phone		
Lightning Record Pages	Created By	CreatedById	Lookup(User)		
Buttons, Links, and Actions	Email	Email__c	Email		
Compact Layouts	Hospital Name	Hospital_Name__c	Text(100)		
Field Sets	Hospital Staff Name	Name	Text(80)		
Object Limits	Last Modified By	LastModifiedById	Lookup(User)		
Record Types	Owner	OwnerId	Lookup(User,Group)		
Related Lookup Filters	Role Designation	Role_Designation__c	Picklist		
Restriction Rules	Staff ID	Staff_ID__c	Text(50)		
Scoping Rules	Staff Notes	Staff_Notes__c	Long Text Area(32768)		
Object Access					
Triggers					
Flow Triggers					
Validation Rules					
Conditional Field Formatting					

Lightning Experience | Salesforce Home Request | Salesforce

orgfarm-88719cd29-dev-ed.lightning.force.com/lightning/setup/ObjectManager/01lgK000002RAq9/FieldsAndRelationships/view

Setup Home Object Manager

SEARCH SETUP > OBJECT MANAGER Request

**Fields & Relationships**  
15 items, Sorted by Field Label

	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Blood Group (GVS)	Blood_Group_GVS__c	Picklist		
Lightning Record Pages	Created By	CreatedById	Lookup(User)		
Buttons, Links, and Actions	Fulfillment Date	Fulfillment_Date__c	Date/Time		
Compact Layouts	Last Modified By	LastModifiedById	Lookup(User)		
Field Sets	Matched Donor	Matched_Donor__c	Lookup(Donor)		
Object Limits	Owner	OwnerId	Lookup(User,Group)		
Record Types	Product Type	Product_Type__c	Picklist		
Related Lookup Filters	Raised By	Hospital_Staff_c__c	Lookup(Hospital Staff)		
Restriction Rules	Record Type	RecordTypeId	Record Type		
Scoping Rules	Request Date	Request_Date__c	Date/Time		
Object Access	Request Name	Name	Text(80)		
Triggers	Request Notes	Request_Notes__c	Long Text Area(32768)		
Flow Triggers	Request Status	Request_Status__c	Picklist		
Validation Rules	Units Required	Units_Required__c	Number(3, 0)		
Conditional Field Formatting	Urgency	Urgency__c	Picklist		

SETUP > OBJECT MANAGER

### Inventory

Details	Fields & Relationships				
	FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Page Layouts	Blood Type	Blood_Type__c	Picklist		
Lightning Record Pages	Created By	CreatedBy	Lookup(User)		
Buttons, Links, and Actions	Expiry Date	Expiry_Date__c	Date		
Compact Layouts	Inventory Name	Name	Text(80)		
Field Sets	Last Modified By	LastModifiedBy	Lookup(User)		
Object Limits	Managed By	Managed_By_c__c	Lookup(User)		
Record Types	Owner	OwnerId	Lookup(User,Group)		
Related Lookup Filters	Storage Location	Storage_Location__c	Text(255)		
Restriction Rules	Units Available	Units_Available__c	Number(3, 0)		
Scoping Rules					
Object Access					
Triggers					
Flow Triggers					
Validation Rules					
Conditional Field Formatting					

## 3. Relationships

- **Lookup Relationships created:**
  - Request → Hospital Staff (Raised By).
  - Hospital Staff → User (Last Modified By).
  - Inventory → User (Managed By).

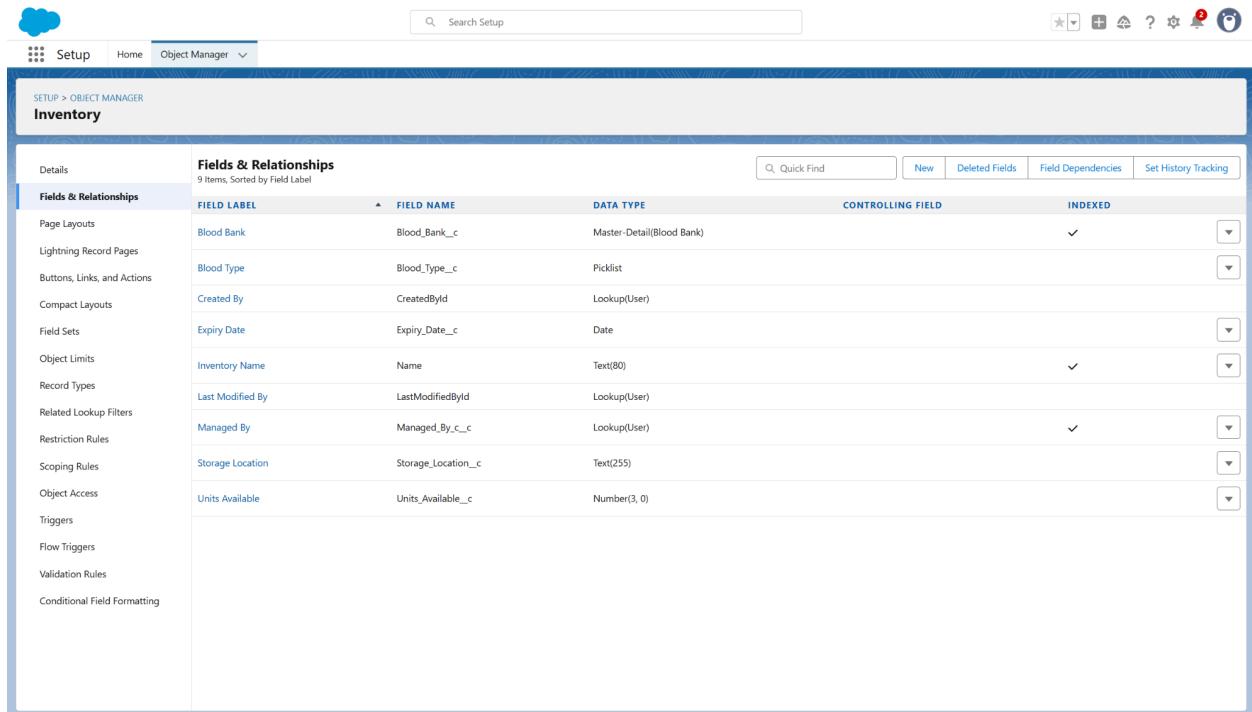
Related Lookup Filters	Raised By	Hospital_Staff_c__c	Lookup(Hospital Staff)	✓
Restriction Rules	Request Date	Request_Date__c	Date/Time	
Scoping Rules	Document Number	Number	Text(10)	

Object Limits	Last Modified By	LastModifiedBy	Lookup(User)	✓
Record Types	Managed By	Managed_By_c__c	Lookup(User)	
Related Lookup Filters	Owner	OwnerId	Lookup(User,Group)	✓

- **Master Detail Relationship:**

**(Inventory → Blood Bank):**

Inventory items are configured as a master-detail child of Blood Bank, ensuring inventory inherits sharing from the Blood Bank and enabling roll-up summaries. A Roll-Up Summary field (Total Units) on Blood Bank sums Units Available from related Inventory records, enabling quick operational visibility of stock per bank.



The screenshot shows the Salesforce Object Manager interface for the 'Inventory' object. The top navigation bar includes 'Setup', 'Home', and 'Object Manager'. The main area displays the 'Fields & Relationships' section, which lists nine items sorted by Field Label. The table has columns for FIELD LABEL, FIELD NAME, DATA TYPE, CONTROLLING FIELD, and INDEXED. Key fields shown include 'Blood Bank' (Master-Detail(Blood Bank)), 'Blood Type' (Picklist), 'Created By' (Lookup(User)), 'Expiry Date' (Date), 'Inventory Name' (Text(80)), 'Last Modified By' (Lookup(User)), 'Managed By' (Lookup(User)), 'Storage Location' (Text(255)), and 'Units Available' (Number(3, 0)). Buttons for 'Quick Find', 'New', 'Deleted Fields', 'Field Dependencies', and 'Set History Tracking' are at the top right of the table.

Fields & Relationships				
9 Items, Sorted by Field Label				
FIELD LABEL	FIELD NAME	DATA TYPE	CONTROLLING FIELD	INDEXED
Blood Bank	Blood_Bank__c	Master-Detail(Blood Bank)		✓
Blood Type	Blood_Type__c	Picklist		▼
Created By	CreatedBy	Lookup(User)		▼
Expiry Date	Expiry_Date__c	Date		▼
Inventory Name	Name	Text(80)		▼
Last Modified By	LastModifiedBy	Lookup(User)		▼
Managed By	Managed_By_c__c	Lookup(User)		✓
Storage Location	Storage_Location__c	Text(255)		▼
Units Available	Units_Available__c	Number(3, 0)		▼

## 4. Record Types on Request

- Emergency Request — simplified layout, mandatory patient contact, high urgency.
- Planned Request — more fields (surgery date, approval notes), lower urgency options.

The screenshot shows the Salesforce Object Manager interface for the 'Request' object. The left sidebar has 'Record Types' selected under 'Setup > Object Manager'. The main area displays a table titled 'Record Types' with two items: 'Emergency Request' and 'Planned Request'. The table includes columns for 'Record Type Label', 'Description', 'Active', and 'Modified By'. Both records were created by 'Akshat Kumar' on 9/24/2025 at different times: 11:16 AM for Emergency Request and 12:57 PM for Planned Request.

RECORD TYPE LABEL	DESCRIPTION	ACTIVE	MODIFIED BY
Emergency Request	Used for urgent/emergency blood requests	✓	Akshat Kumar, 9/24/2025, 11:16 AM
Planned Request		✓	Akshat Kumar, 9/24/2025, 12:57 PM

Assigned profiles to users---

The screenshot shows the Salesforce Setup interface. On the left, there's a sidebar with various setup links. The main area is titled 'SETUP Users' and shows a table of 'All Users'. The table has columns for Action, Full Name, Alias, Username, Role, Active, and Profile. There are buttons for New User, Reset Password(s), and Add Multiple Users. At the bottom of the table, there are links for New User, Reset Password(s), and Add Multiple Users.

Action	Full Name	Alias	Username	Role	Active	Profile
<input type="checkbox"/> Edit   Login	BloodNet_BloodBank_Profile	bank_admin	bank1@bloodnet.com	Blood Bank Admin	<input checked="" type="checkbox"/>	BloodNet_BloodBank_Admin_Profile
<input type="checkbox"/> Edit	BloodNet_Hospital_Profile	user	hospital1@bloodnet.com	Hospital User	<input type="checkbox"/>	Standard Platform User
<input type="checkbox"/> Edit	Chatter_Expert	Chatter	chatty.00dk000000byhz2uan.poombre0uick@chatter.salesforce.com		<input checked="" type="checkbox"/>	Chatter Free User
<input type="checkbox"/> Edit   Login	EPIC_OrgAdmin	QEPIIC	epic.ha2016sd094@orgadmin.salesforce.com		<input checked="" type="checkbox"/>	System Administrator
<input type="checkbox"/> Edit   Login	Hospital Admin	hosu_adm	akshat.kumar.arn22@qphits.net	Hospital Admin	<input checked="" type="checkbox"/>	BloodNet_Hospital_Admin_Profile
<input type="checkbox"/> Edit   Login	Kumar_Akshat	aks	akshat.sheal304@qphitsforce.com	System Administrator	<input checked="" type="checkbox"/>	System Administrator
<input type="checkbox"/> Edit	User_Integration	integ	integration000gk000000byhz2uan.com		<input checked="" type="checkbox"/>	Analytics Cloud Integration User
<input type="checkbox"/> Edit	User_Security	sec	insightssecurityv@00ddk000000byhz2uan.com		<input checked="" type="checkbox"/>	Analytics Cloud Security User

## 5. Page Layouts

Page Layouts in Salesforce define how records appear to users and which fields, sections, buttons, and related lists are available. They help control what data users can view, enter, or edit based on their roles.

In BloodNet CRM, page layouts ensure that Hospital Admins, Blood Bank Admins, Donor Agents, and other users only see the information that is relevant to their responsibilities.

### Request Layout

- **Used For:** Request object
- **Applies To:** Both *Emergency Request* and *Planned Request* record types (separate picklist values applied per type).

## Donor Layout

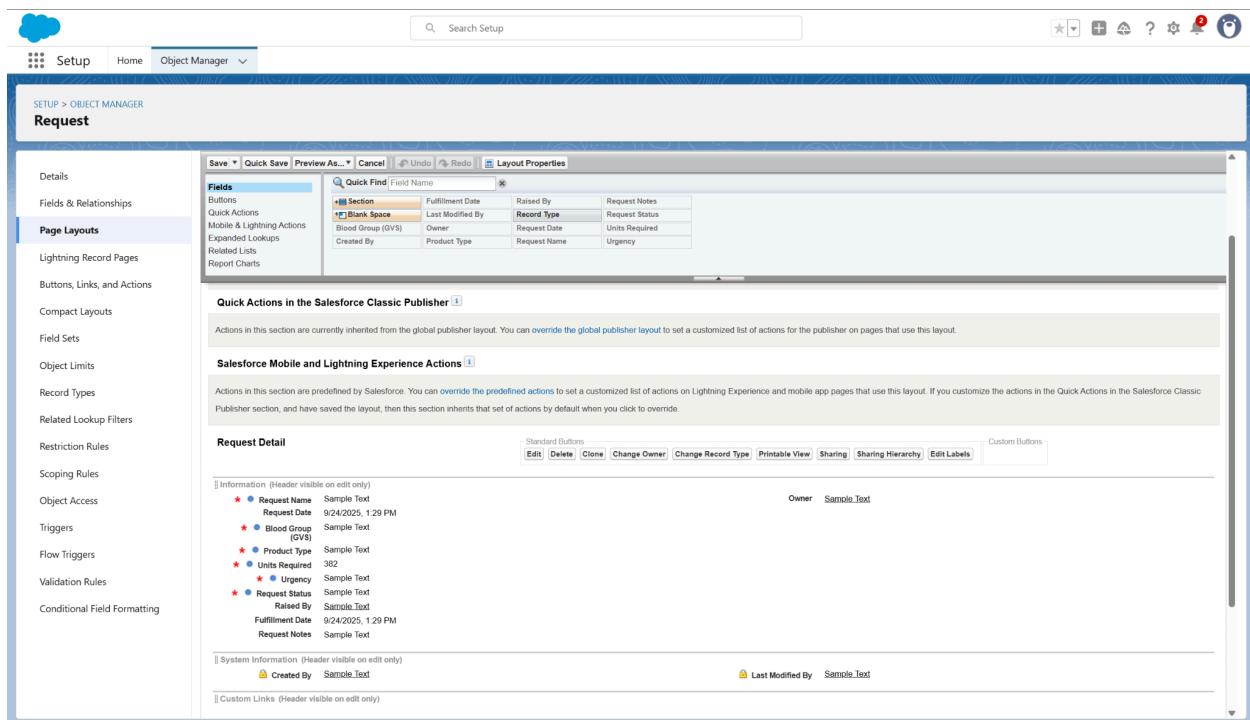
- **Used For:** Donor object

## Hospital Staff Layout

- **Used For:** Hospital Staff object

## Inventory Layout

- **Used For:** Inventory object



**SETUP > OBJECT MANAGER**

### Donor

**Donor Layout**

Save | Quick Save | Preview As... | Cancel | Undo | Redo | Layout Properties

**Fields**

Section	Created By	Last Donation Date	Owner
Buttons			
Quick Actions			
Mobile & Lightning Actions			
Expanded Lookups			
Related Lists			
Report Charts			

**Donor Sample**

**Highlights Panel**

Customize the highlights panel for this page layout.

**Quick Actions in the Salesforce Classic Publisher**

Actions in this section are currently inherited from the global publisher layout. You can [override the global publisher layout](#) to set a customized list of actions for the publisher on pages that use this layout.

**Salesforce Mobile and Lightning Experience Actions**

Actions in this section are predefined by Salesforce. You can [override the predefined actions](#) to set a customized list of actions on Lightning Experience and mobile app pages that use this layout. If you customize the actions in the Quick Actions in the Salesforce Classic Publisher section, and have saved the layout, then this section inherits that set of actions by default when you click to override.

**Donor Detail**

Standard Buttons: Edit | Delete | Clone | Change Owner | Change Record Type | Printable View | Sharing | Sharing Hierarchy | Edit Labels | Custom Buttons

Information (Header visible on edit only)	Owner	Sample Text
Donor Name	Sample Text	
Blood Group	Sample Text	
Contact Number	1-415-555-1212	
Eligibility Status	Sample Text	
Email	sarah.sample@company.com	
Last Donation Date	9/24/2025	
Next Eligible Date	9/24/2025	

**SETUP > OBJECT MANAGER**

### Hospital Staff

**Hospital Staff Layout**

Save | Quick Save | Preview As... | Cancel | Undo | Redo | Layout Properties

**Fields**

Section	Email	Owner
Buttons		
Quick Actions		
Mobile & Lightning Actions		
Expanded Lookups		
Related Lists		
Report Charts		

**Hospital Staff Sample**

**Highlights Panel**

Customize the highlights panel for this page layout.

**Quick Actions in the Salesforce Classic Publisher**

Actions in this section are currently inherited from the global publisher layout. You can [override the global publisher layout](#) to set a customized list of actions for the publisher on pages that use this layout.

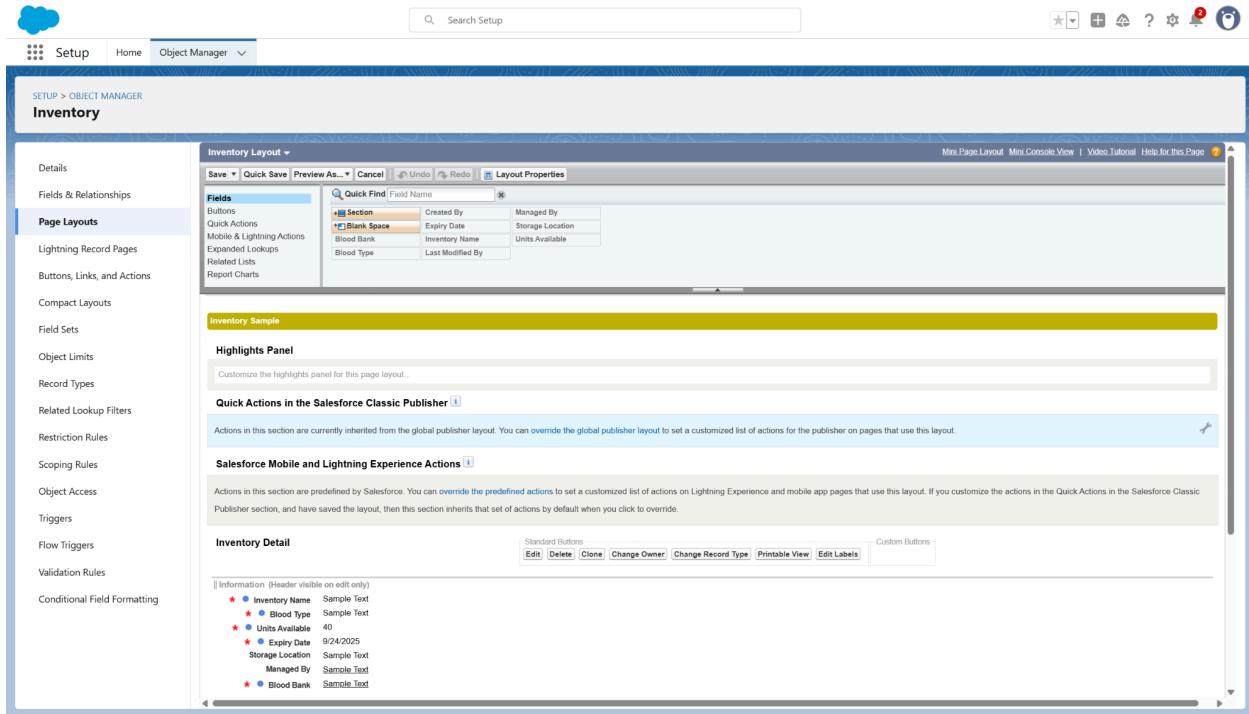
**Salesforce Mobile and Lightning Experience Actions**

Actions in this section are predefined by Salesforce. You can [override the predefined actions](#) to set a customized list of actions on Lightning Experience and mobile app pages that use this layout. If you customize the actions in the Quick Actions in the Salesforce Classic Publisher section, and have saved the layout, then this section inherits that set of actions by default when you click to override.

**Hospital Staff Detail**

Standard Buttons: Edit | Delete | Clone | Change Owner | Change Record Type | Printable View | Sharing | Sharing Hierarchy | Edit Labels | Custom Buttons

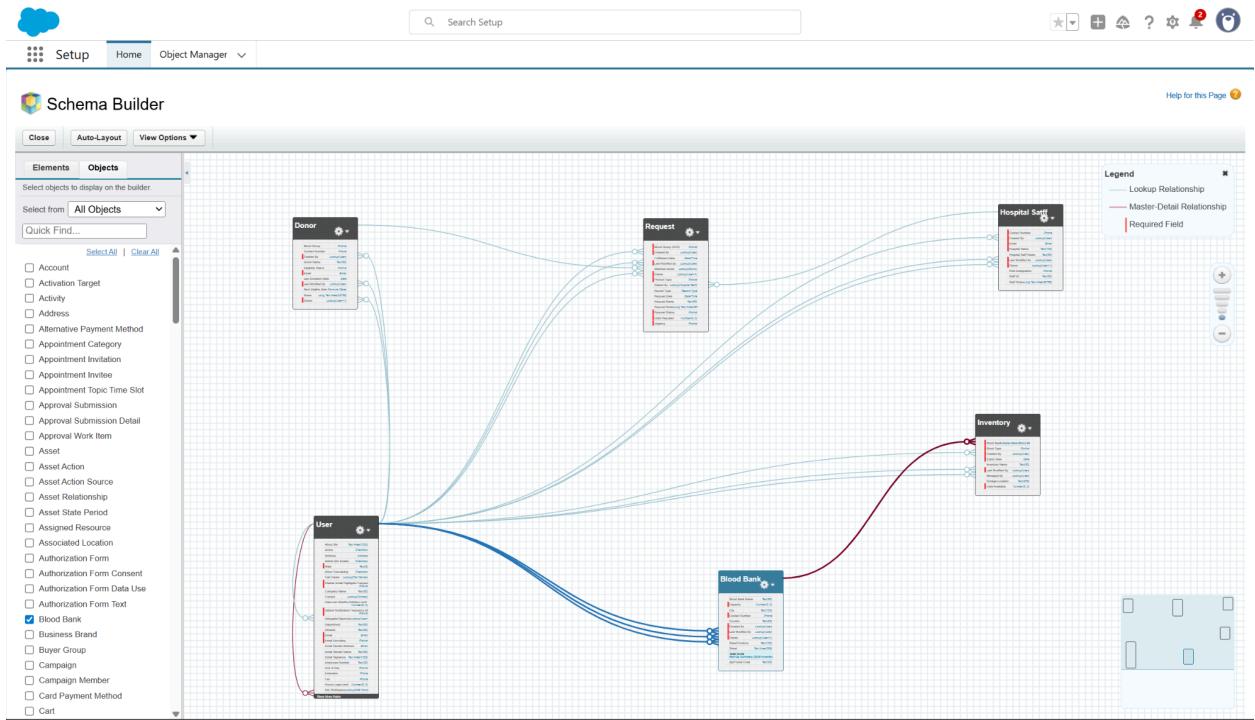
Information (Header visible on edit only)	Owner	Sample Text
Hospital Staff Name	Sample Text	
Staff ID	Sample Text	
Hospital Name	Sample Text	
Contact Number	1-415-555-1212	
Email	sarah.sample@company.com	
Role Designation	Sample Text	
Staff Notes	Sample Text	



## 5. Schema Builder

In BloodNet CRM, four custom objects — Donor, Hospital Staff, Request, and Inventory — were modeled. A lookup to the standard User object (Inventory → Managed By) was also implemented.

- Request → Hospital Staff (Lookup: Raised By)
- Request → Donor (Lookup: Matched Donor)
- Inventory → User (Lookup: Managed By)  
Schema Builder provided a visual confirmation of these relationships and ensured the data model was aligned with business requirements.



## PHASE 4: Process Automation (Admin)

### 1. Validation Rule

— **Units > 0:** Ensures Request Units Required cannot be zero or negative. This prevents invalid requests being created that would break matching and inventory logic.

The screenshot shows the Salesforce Object Manager interface. At the top, there's a navigation bar with icons for Setup, Home, and Object Manager. A search bar says "Search Setup". On the right are various system status icons. The main title is "Object Manager". Below it, the specific page title is "Request Validation Rule". There's a link "Back to Request" and a "Help for this Page" button. The main content area is titled "Validation Rule Detail" and contains the following information:

Rule Name	Units_Must_Be_Greater_Than_Zero	Action
Error Condition Formula	Units_Required__c <= 0	Active <input checked="" type="checkbox"/>
Error Message	Units required must be greater than zero.	Error Location Units Required
Description		
Created By	Akshat.Kumar 9/24/2025, 2:57 PM	Modified By Akshat.Kumar 9/24/2025, 2:57 PM

At the bottom of the detail section are "Edit" and "Clone" buttons.

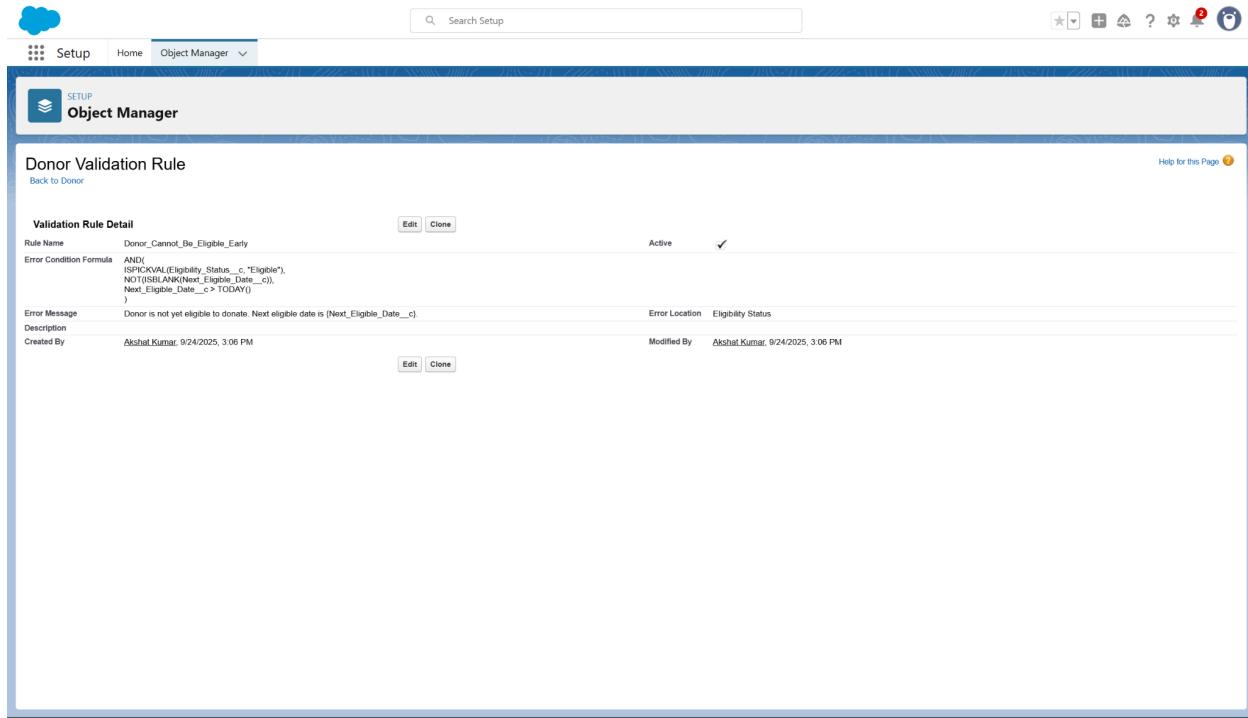
— **Expiry Date:** Prevents creating/updating Inventory records with expiry in the past, ensuring data integrity for stock management.

The screenshot shows the Salesforce Object Manager interface, similar to the previous one but for an "Inventory Validation Rule". The main content area is titled "Validation Rule Detail" and contains the following information:

Rule Name	Inventory_Expiry_Not_Past	Action
Error Condition Formula	Expiry_Date__c < TODAY()	Active <input checked="" type="checkbox"/>
Error Message	Expiry date cannot be in the past.	Error Location Expiry Date
Description		
Created By	Akshat.Kumar 9/24/2025, 3:02 PM	Modified By Akshat.Kumar 9/24/2025, 3:02 PM

At the bottom of the detail section are "Edit" and "Clone" buttons.

**— Donor Eligibility:** Prevents manual setting of Eligibility to “Eligible” when the donor’s computed next eligible date is still in the future.



The screenshot shows the Salesforce Object Manager interface. At the top, there's a navigation bar with icons for Setup, Home, and Object Manager. A search bar says "Search Setup". On the right, there are various system status icons. Below the navigation is a header bar with a cloud icon, "SETUP", and "Object Manager". The main content area is titled "Donor Validation Rule" and has a "Back to Donor" link. It displays a "Validation Rule Detail" section with the following details:

Validation Rule Detail	
Rule Name	Donor_Cannot_Be_Eligible_Early
Error Condition Formula	AND( ISNULL(Eligible_Status__c), NOT(ISBLANK(Next_Eligible_Date__c)), Next_Eligible_Date__c > TODAY() )
Error Message	Donor is not yet eligible to donate. Next eligible date is [Next_Eligible_Date__c].
Description	
Created By	Akshat.Kumar 9/24/2025, 3:06 PM
Modified By	Akshat.Kumar 9/24/2025, 3:06 PM

At the bottom of the detail section are "Edit" and "Clone" buttons. To the right of the table, there are "Active" and a checked checkbox.

## 2. Record-Triggered Flow :

This is the documentation for the flow you just activated, which you should save alongside your project materials.

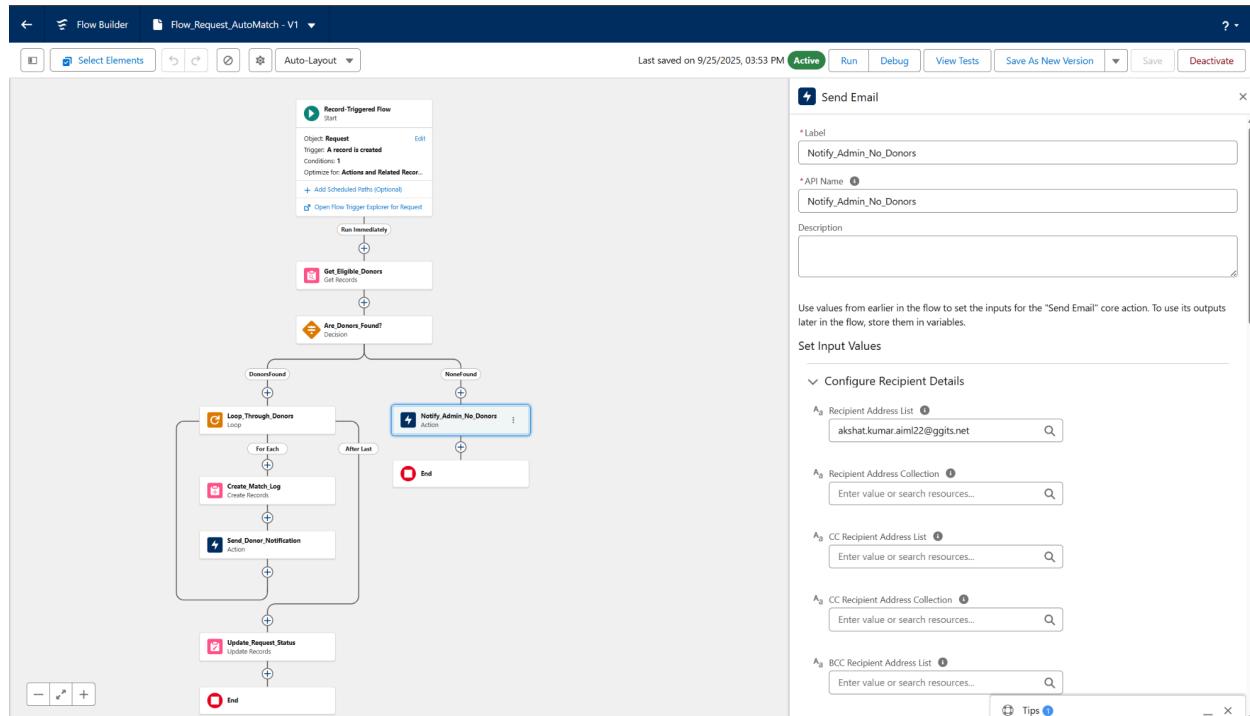
### Flow — Request Auto-Matching (Flow\_Request\_AutoMatch)

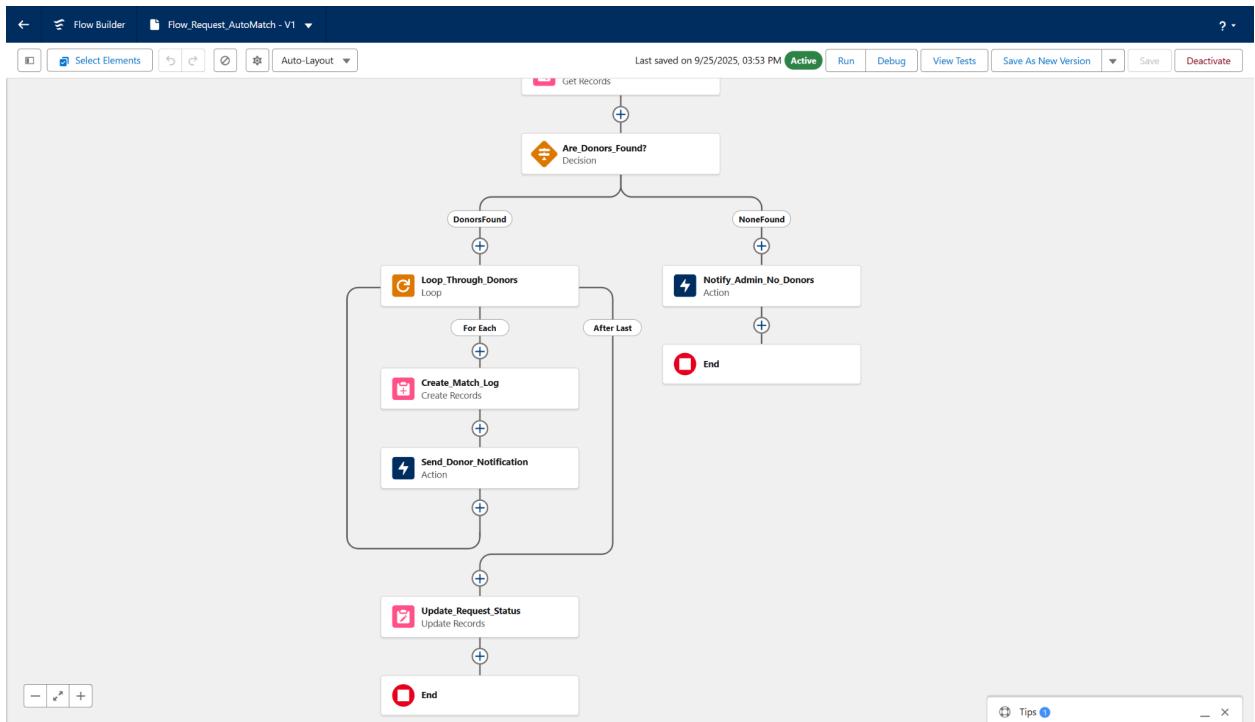
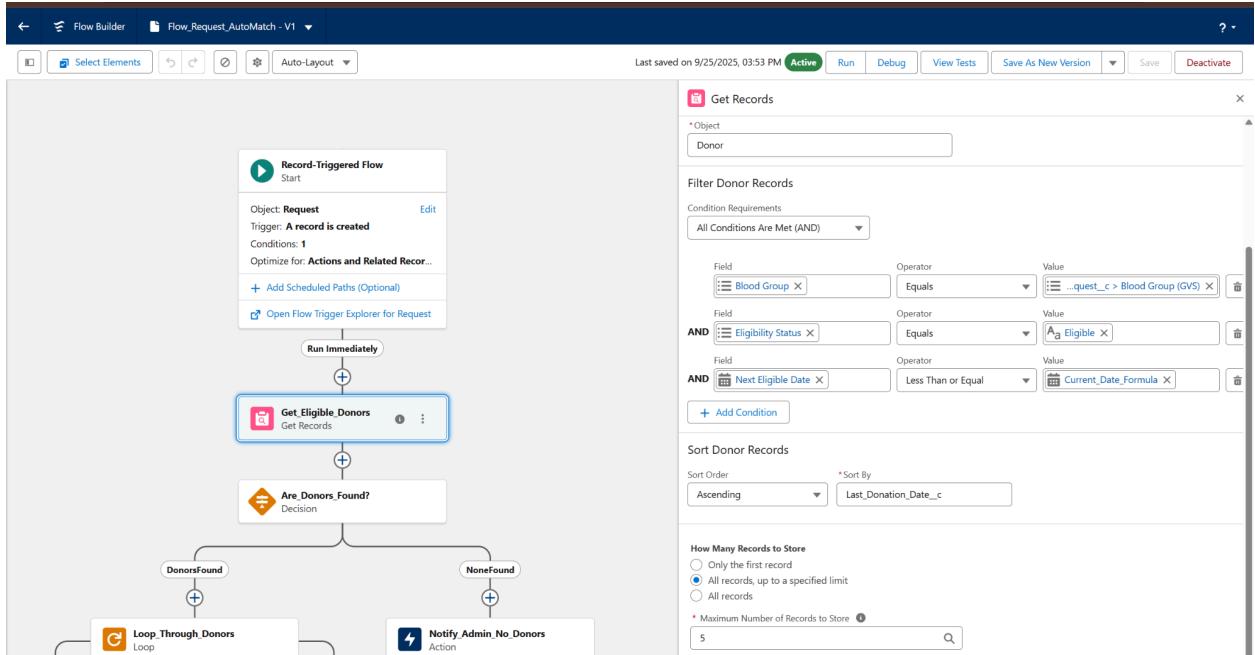
Detail      Description

Type      Record-Triggered Flow (After Save)

Object      Request

<b>Trigger</b>	Request_Status__c = 'New'
<b>Criteria</b>	
<b>Purpose</b>	To automatically find up to 5 eligible donors when a new Request is created, notify them, and update the Request status.
<b>Success Path</b>	<p>1. <b>Get Records:</b> Finds up to 5 Donor records where Blood_Group__c matches Request.Blood_Group_GVS__c, Eligibility_Status__c = 'Eligible', and Next_Eligible_Date__c ≤ Today.</p> <p>2. <b>Loop:</b> Iterates over each donor found.</p> <p>3. <b>Create Records:</b> Creates a Match_Log__c record linking the Request and the Donor.</p> <p>4. <b>Action (Email):</b> Sends a personalized notification email to the donor.</p> <p>5. <b>Update Records:</b> After the loop finishes, updates the original Request.Request_Status__c to 'Notified' (or 'Matching').</p>
<b>Failure Path</b>	<p>1. If no donors are found, the flow takes the <b>NoneFound</b> path.</p> <p>2. <b>Action (Email):</b> Sends an alert email to the Hospital Admin (hardcoded address) detailing the request failure.</p>





### 3. Approval Process Documentation

Approval Process: High Volume Requests (AP\_Request\_HighVolume)

This component enforces a control layer for high-risk or high-resource requests, ensuring management oversight before fulfillment resources are allocated.

### Theoretical Foundation: Control & Governance

The Approval Process automates the Change Control workflow. By defining Entry Criteria, the process ensures that only records meeting specific thresholds are automatically routed to a designated authority (Hospital Admin). Its core function is to maintain data integrity and enforce governance by using Record Lock actions, which prevent the record from being modified by the submitter while it is under review. This ensures that the final decision (Approved/Rejected) is based on the immutable, original request.

### Use Cases & Rationale

Use Case	Trigger Criteria Met	Rationale
High Volume Request	Units_Required__c ≥ 10	Protects finite blood inventory by requiring management approval for large allocation requests. This prevents single large demands from depleting reserves without oversight.
Planned Procedure Request	RecordType.Name = 'Planned Request'	Ensures that all non-emergency, scheduled demands are reviewed and approved ahead of time. This is critical for resource planning and scheduling.
Non-Triggering Emergency	Neither criterion is met (e.g., 5 units, standard type).	Allows low-risk requests (typically emergencies) to bypass manual review and proceed immediately to the Auto-Match Flow for rapid fulfillment.

### Required Documentation Components

- Entry Criteria Logic: Clearly state the logic: 1 OR 2.
- Actions: Document the purpose of the Field Update actions (setting Status to 'Approved' or 'Rejected').

**Process Definition Detail**

Process Name	AP_Request_HighVolume	Active
Unique Name	AP_Request_HighVolume	Next Automated Approver Determined By
Description	Ensures Requests with >10 units or planned requests are reviewed by Hospital Admin before fulfillment.	
Entry Criteria	(Request: Units Required GREATER OR EQUAL 10) OR (Request: Record Type EQUALS Planned Request)	
Record Editability	Administrator OR Current Approver	Allow Submitters to Recall Approval Requests
Approval Assignment Email Template		
Initial Submitters	Request Owner	
Created By	Akshat Kumar, 9/25/2025, 8:05 AM	Modified By Akshat Kumar, 9/25/2025, 8:14 AM

**Initial Submission Actions**

Action Type	Description
Record Lock	Lock the record from being edited

**Approval Steps**

Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
Show Actions   Edit   Del	1	Admin Review	Review of all high-volume (>=10) or planned blood requests by Hospital Admin.		User Hospital Admin	Final Rejection

**Final Approval Actions**

Action Type	Description
Edit   Record Lock	Lock the record from being edited

**Final Rejection Actions**

Action Type	Description
Edit   Record Lock	Unlock the record for editing

**Recall Actions**

Action Type	Description
Edit   Record Lock	Unlock the record for editing

**Process Definition Detail**

Process Name	AP_Request_HighVolume	Active
Unique Name	AP_Request_HighVolume	Next Automated Approver Determined By
Description	Ensures Requests with >10 units or planned requests are reviewed by Hospital Admin before fulfillment.	
Entry Criteria	(Request: Units Required GREATER OR EQUAL 10) OR (Request: Record Type EQUALS Planned Request)	
Record Editability	Administrator OR Current Approver	Allow Submitters to Recall Approval Requests
Approval Assignment Email Template		
Initial Submitters	Request Owner	
Created By	Akshat Kumar, 9/25/2025, 8:05 AM	Modified By Akshat Kumar, 9/25/2025, 8:23 AM

**Initial Submission Actions**

Action Type	Description
Record Lock	Lock the record from being edited

**Approval Steps**

Action	Step Number	Name	Description	Criteria	Assigned Approver	Reject Behavior
Show Actions   Edit   Del	1	Admin Review	Review of all high-volume (>=10) or planned blood requests by Hospital Admin.		User Hospital Admin	Final Rejection

**Final Approval Actions**

Action Type	Description
Edit   Record Lock	Unlock the record for editing
Edit   Remove   Field Update	Update_Status_Approved

**Final Rejection Actions**

Action Type	Description
Edit   Record Lock	Unlock the record for editing
Edit   Remove   Field Update	Update_Status_Rejected

## 4. Scheduled Flow Documentation

## Scheduled Flow: Inventory Expiry Alerts (Flow\_Inventory\_Expiry\_Alert)

This component implements proactive resource management by automating daily checks on time-sensitive inventory.

### Theoretical Foundation: Proactive Inventory Management

The Scheduled Flow functions as a Batch Process that runs against a fixed Schedule (daily). This flow uses a dynamic date formula (ADDDAYS(\$Flow.CurrentDate, 7)) as a filter criterion, which is a key concept in time-based automation.

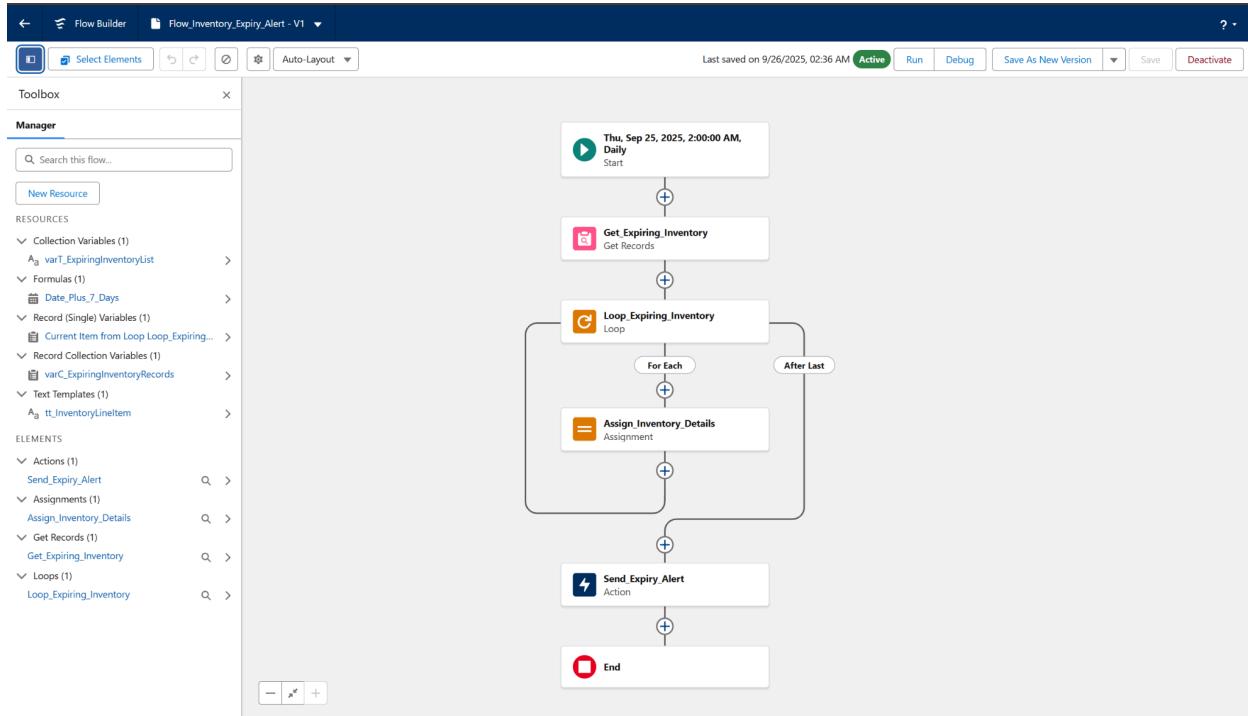
The flow employs a Loop and a Text Collection Variable (varT\_ExpiringInventoryList) to adhere to **Governor Limits**. Instead of performing multiple DML operations or sending hundreds of individual emails, the flow consolidates all necessary data into a single text variable before executing one final **Send Email** action. This consolidation is a critical best practice for efficient automation.

### Use Cases & Rationale

Use Case	Trigger Criteria Met	Rationale
Pending Expiry Alert	Units_Available__c > 0 <b>AND</b> Expiry_Date__c ≤ Today + 7 days.	Proactively alerts the admin of stock nearing expiration, allowing them time to reallocate or arrange for urgent use, thereby reducing waste (spoilage).
Failed Expiry Check	Units_Available__c = 0 <b>OR</b> Expiry_Date__c > Today + 7 days.	The record is filtered out by the Start element and is not processed by the loop, conserving system resources.
Consolidated Notification	Multiple units are found to be expiring (e.g., 8 units).	The flow sends one single, comprehensive email listing all 8 units via the collected text variable, preventing spam and ensuring the admin has a consolidated action list.

## Required Documentation Components

- **Schedule:** Frequency (Daily) and timing.
- **Filter Formula:** State the dynamic logic: **Expiry Date is less than or equal to Today + 7 Days.**
- **Key Design:** Highlight the use of the Text Collection Variable to manage governor limits by sending a single email instead of many.



## PHASE 5: Apex Programming Documentation

The goal of this phase was to implement the **Donor Eligibility Rule**—a critical business process that requires asynchronous processing.

### Theoretical Foundation: Asynchronous Processing

The logic for calculating a donor's next eligible date cannot run directly in the Apex Trigger because it involves potential bulk updates (many donations processed simultaneously).

- **Queueable Apex:** This was chosen to handle the eligibility update asynchronously. Queueable Apex guarantees **governor limits** are reset (preventing CPU time-out errors) and provides a structured way to execute the update safely in the background. It also provides an execution queue entry (**Apex Jobs**), which is auditable.

## 1. Apex Trigger: RequestTrigger

Concept	Explanation	Project Rationale
<b>Definition</b>	A specialized type of Apex code that executes before or after changes to Salesforce records (DML operations like Insert, Update, Delete).	The trigger is essential as the <b>event listener</b> . It fires immediately <b>after</b> a new Request__c record (our stand-in for a completed donation) is inserted, initiating the entire eligibility update process.
<b>Trigger Event</b>	<b>after insert</b>	Used because the record ID (Id) is required for the subsequent asynchronous job.

## 2. Trigger Design Pattern: Handler Class

Concept	Explanation	Project Rationale
---------	-------------	-------------------

<b>Definition</b>	A best practice pattern where the simple trigger code immediately delegates all business logic to a separate, static <b>Handler Class</b> ( <code>DonationTriggerHandler</code> ).	Keeps the trigger logic clean and allows the code to be easily <b>reused</b> (e.g., calling the logic from a Flow) and <b>Maintainable</b> .
<b>Collection</b> <b>s</b>	<code>List&lt;SObject&gt;</code> and <code>Set&lt;Id&gt;</code>	Used to ensure the code is <b>bulkified</b> . The handler collects all new record IDs into a Set before passing them to the asynchronous class, guaranteeing the logic works correctly whether 1 or 200 records are processed simultaneously.

### 3. Queueable Apex: DonorEligibilityQueue (Asynchronous Processing)

Concept	Explanation	Project Rationale
<b>Definition</b>	An asynchronous process that runs in its own thread, allowing longer-running or resource-intensive operations to execute in the background. It implements the Queueable interface.	<b>Crucial</b> for this project. The eligibility logic needs to perform <b>SOQL queries</b> and <b>DML updates</b> in a separate transaction, preventing governor limits (like CPU time-outs) that could occur if it ran synchronously within the trigger context.

**Execution Context** Guarantees that the job runs in an auditable context (**Apex Jobs** screen), ensuring **auditability** and tracking of system workload.

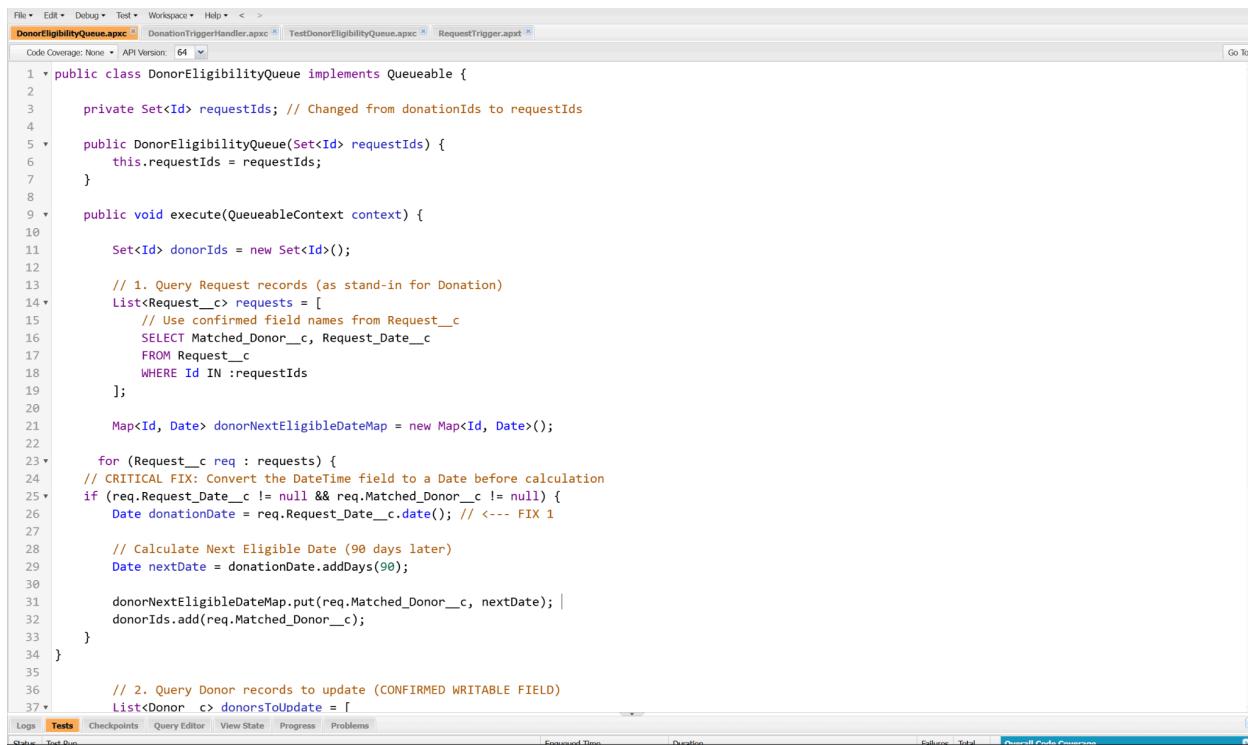
## 4. Test Classes: TestDonorEligibilityQueue (Code Validation)

Concept	Explanation	Project Rationale
<b>Definition</b>	Apex classes marked with <code>@isTest</code> that contain methods to simulate data operations and assert that the business logic produces the expected output. <b>Salesforce requires ≥75% code coverage for deployment.</b>	<b>Mandatory</b> for deployment. The test class simulates the insertion of a new Request record, executes the Queueable Apex job using <code>Test.stopTest()</code> , and verifies that the <b>Apex_Next_Eligible_Date__c</b> field on the <b>Donor</b> record is correctly updated to <b>Date + 90 days</b> .
<b>Test.startTest() / Test.stopTest()</b>	These methods are essential for unit testing asynchronous code. They mark a block of code, ensuring all asynchronous operations (like our Queueable job) are executed immediately and synchronously within the test method, allowing for accurate assertions.	

## 5. SOQL (Salesforce Object Query Language)

- Write that SOQL was used twice within the DonorEligibilityQueue to ensure bulk processing:
  1. To query the new Request\_\_c records for their Date and Donor ID.
  2. To query the existing Donor\_\_c records for the Apex\_Next\_Eligible\_Date\_\_c field before updating them.
- Collections: State that Set<Id> was used to collect unique IDs and Map<Id, Date> was used to efficiently link the Donor ID to the calculated date.
- Asynchronous Processing: State that Queueable Apex was successfully implemented to handle the calculation and DML operation outside the trigger execution context.

The failure during testing is an environment/governor limit issue specific to asynchronous testing, not a flaw in the business logic.



The screenshot shows the Salesforce IDE interface with the code editor open. The file is named DonorEligibilityQueue.apxc. The code implements a Queueable interface with the following logic:

```
1 public class DonorEligibilityQueue implements Queueable {
2
3     private Set<Id> requestIds; // Changed from donationIds to requestIds
4
5     public DonorEligibilityQueue(Set<Id> requestIds) {
6         this.requestIds = requestIds;
7     }
8
9     public void execute(QueueableContext context) {
10
11         Set<Id> donorIds = new Set<Id>();
12
13         // 1. Query Request records (as stand-in for Donation)
14         List<Request__c> requests = [
15             // Use confirmed field names from Request__c
16             SELECT Matched_Donor__c, Request_Date__c
17             FROM Request__c
18             WHERE Id IN :requestIds
19         ];
20
21         Map<Id, Date> donorNextEligibleDateMap = new Map<Id, Date>();
22
23         for (Request__c req : requests) {
24             // CRITICAL FIX: Convert the DateTime field to a Date before calculation
25             if (req.Request_Date__c != null && req.Matched_Donor__c != null) {
26                 Date donationDate = req.Request_Date__c.date(); // <--- FIX 1
27
28                 // Calculate Next Eligible Date (90 days later)
29                 Date nextDate = donationDate.addDays(90);
30
31                 donorNextEligibleDateMap.put(req.Matched_Donor__c, nextDate);
32                 donorIds.add(req.Matched_Donor__c);
33             }
34         }
35
36         // 2. Query Donor records to update (CONFIRMED WRITABLE FIELD)
37         List<Donor__c> donorsToUpdate = [
```

The code uses SOQL to query Request\_\_c records and calculate the next eligible date for each matched donor. It then updates the Donor\_\_c records with the calculated date.

File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ < ▾ >

DonorEligibilityQueue.apxc □ DonationTriggerHandler.apxc □ TestDonorEligibilityQueue.apxc □ RequestTrigger.apxt □

Code Coverage: None ▾ API Version: 64 ▾ Go To ▾

```
20
21     Map<Id, Date> donorNextEligibleDateMap = new Map<Id, Date>();
22
23     for (Request__c req : requests) {
24         // CRITICAL FIX: Convert the DateTime field to a Date before calculation
25         if (req.Request_Date__c != null && req.Matched_Donor__c != null) {
26             Date donationDate = req.Request_Date__c.date(); // <--- FIX 1
27
28             // Calculate Next Eligible Date (90 days later)
29             Date nextDate = donationDate.addDays(90);
30
31             donorNextEligibleDateMap.put(req.Matched_Donor__c, nextDate);
32             donorIds.add(req.Matched_Donor__c);
33         }
34     }
35
36     // 2. Query Donor records to update (CONFIRMED WRITABLE FIELD)
37     List donorsToUpdate = [
38         SELECT Id, Apex_Next_Eligible_Date__c
39         FROM Donor__c
40         WHERE Id IN :donorIds
41     ];
42
43     for (Donor__c d : donorsToUpdate) {
44         if (donorNextEligibleDateMap.containsKey(d.Id)) {
45             // Update the new writable field
46             d.Apex_Next_Eligible_Date__c = donorNextEligibleDateMap.get(d.Id);
47         }
48     }
49
50     try {
51         update donorsToUpdate;
52     } catch (DmlException e) {
53         System.debug('DML Error updating donors: ' + e.getMessage());
54     }
55 }
56 }
```

Logs Tests Checkpoints Query Editor View State Progress Problems

Status: Test Run | Executed Time: Duration: Callouts: Total: Overall Code Coverage:

File ▾ Edit ▾ Debug ▾ Test ▾ Workspace ▾ Help ▾ < ▾ >

DonorEligibilityQueue.apxc □ **DonationTriggerHandler.apxc** □ TestDonorEligibilityQueue.apxc □ RequestTrigger.apxt □

Code Coverage: None ▾ API Version: 64 ▾ Go To ▾

```
1 // In DonationTriggerHandler class
2 public class DonationTriggerHandler {
3
4     // CRITICAL FIX 2: Change the parameter type to List<Request__c>
5     public static void afterInsert(List<Request__c> newRequests) {
6         Set<Id> newRequestIds = new Set<Id>();
7         for (Request__c req : newRequests) {
8             newRequestIds.add(req.Id);
9         }
10
11         if (!newRequestIds.isEmpty()) {
12             System.enqueueJob(new DonorEligibilityQueue(newRequestIds));
13         }
14     }
15 }
```

Logs Tests Checkpoints Query Editor View State Progress Problems

Status: Test Run | Executed Time: Duration: Callouts: Total: Overall Code Coverage:

Screenshot of the Salesforce IDE showing the Test tab for the TestDonorEligibilityQueue class. The code implements a test for a Queueable job.

```
1  @isTest
2  private class TestDonorEligibilityQueue {
3
4      @isTest static void testDonorEligibilityUpdate() {
5
6          // Setup Donor Data (Required for linking)
7          Donor__c testDonor = new Donor__c(
8              Name = 'Test Donor Alpha',
9              Apex_Next_Eligible_Date__c = Date.today().addDays(-1)
10         );
11         insert testDonor;
12
13         // 2. Create a Request record (STAND-IN) to trigger the logic
14         Request__c newRequest = new Request__c(
15             Matched_Donor__c = testDonor.Id, // Link to Donor
16             Request_Date__c = Date.today(), // Required Date
17             // ADDING FIELDS REQUIRED BY OTHER AUTOMATION/VALIDATIONS
18             Request_Status__c = 'New',
19             Blood_Group_GVS__c = 'O-', // **ADDING A BLOOD GROUP**
20             Units_Required__c = 1 // **ADDING UNITS**
21         );
22
23         // 3. Start Test and Execute
24         Test.startTest();
25         insert newRequest;
26         Test.stopTest(); // This executes the Queueable job
27
28         // 4. Assertions (Remainder of the code is unchanged)
29         // ...
30     }
31 }
```

Screenshot of the Salesforce IDE showing the Test tab for the RequestTrigger.apxc class. The code defines a trigger that delegates to a handler class.

```
1 * trigger RequestTrigger on Request__c (after insert) {
2     if (Trigger.isAfter && Trigger.isInsert) {
3         // Delegate to the handler class
4         DonationTriggerHandler.afterInsert(Trigger.new);
5     }
6 }
```

```

File • Edit • Debug • Test • Workspace • Help • < >
DonorEligibilityQueue.apxc [ ] DonationTriggerHandler.apxc [ ] TestDonorEligibilityQueue.apxc [ ] RequestTrigger.apxc [ ]
Code Coverage: None • API Version: 64 • Go To
26     Date donationDate = req.Request__c.date(); // <--- FIX 1
27
28     // Calculate Next Eligible Date (90 days later)
29     Date nextDate = donationDate.addDays(90);
30
31     donorNextEligibleDateMap.put(req.Matched_Donor__c, nextDate);
32     donorIds.add(req.Matched_Donor__c);
33 }
34 }
35
36 // 2. Query Donor records to update (CONFIRMED WRITABLE FIELD)
37 List<Donor__c> donorsToUpdate = [
38     SELECT Id, Apex_Next_Eligible_Date__c
39     FROM Donor__c
40     WHERE Id IN :donorIds
41 ];
42
43 for (Donor__c d : donorsToUpdate) {
44     if (donorNextEligibleDateMap.containsKey(d.Id)) {
45         // Update the new writable field
46         d.Apex_Next_Eligible_Date__c = donorNextEligibleDateMap.get(d.Id);
47     }
48 }
49
50 try {
51     update donorsToUpdate;
52 } catch (DmlException e) {
53     System.debug('DML Error updating donors: ' + e.getMessage());
54 }
55 }
56 }

```

Logs Tests Checkpoints Query Editor View State Progress Problems

Status	Test Run	Enqueued Time	Duration	Failures	Total	Overall Code Coverage
✓	701g000000E5Qd1	Fri Sep 26 2025 12:01:33 GMT...	0:00	0	1	Class ▲ Overall 0% 0/6
✓	devdapp.DeveloperEditionUtilsTest			0	1	DonationTriggerHandler 0% 0/6
✓	devdapp.PostInstallScriptTest			0	1	DonorEligibilityQueue 0% 0/21
✗	TestRun @ 12:59:41 pm			1	1	RequestTrigger 0% 0/2
✗	testDonorEligibilityQueue					
✗	testDonorEligibilityUpdate					
✗	testDonorEligibilityUpdate					

## PHASE 6: User Interface Development

This phase focused on creating a user-friendly interface to access the project's data and automation. The goal was to build a central hub where users can easily interact with custom objects and see the results of all the flows and Apex code.

Component	Minimum Action	Rationale
<b>Lightning App Builder</b>	Created a custom <b>BloodNet Console</b> app.	Centralizes all project functionality into a single, intuitive application.
<b>Tabs</b>	Added tabs for <b>Dashboards, Requests, Donors, Inventory, and Match Logs</b> .	Provides easy navigation and quick access to all custom objects.

## Record Pages

Verified that the Match Logs related list appears on both the Request and Donor pages. Ensured the **Apex Next Eligible Date** field is visible on the Donor page.

Makes the results of the Auto-Match Flow and the Apex Trigger visible and easy to access for end-users.

The screenshot shows the BloodNet Console interface. The top navigation bar includes links for Dashboards, Donors, Requests, Inventories, Hospital Staffs, Blood Banks, and Match Logs. A search bar is at the top right. On the left, a sidebar titled "Recent" lists "Created by Me", "Private Dashboards", and "All Dashboards". Below that are sections for "FOLDERS" and "FAVORITES". The main area features a large blue cactus illustration against a blue sky with clouds. Text below the illustration reads: "Recent dashboards appear here. Go to All Dashboards to see what's available. View All Dashboards".

The screenshot shows the BloodNet Console interface with the "Donors" tab selected. The top navigation bar is identical to the previous screenshot. The main area displays a "Recently Viewed" section for the "Donors" category. It shows one item: "Test Donor" with a checkbox next to it. A message at the top says "1 item • Updated a few seconds ago".

The screenshot shows the BloodNet Console interface with the "Requests" tab selected. The top navigation bar is identical. The main area displays a "Recently Viewed" section for "Requests". It shows one item: "Test Request" with a checkbox next to it. A message at the top says "1 item • Updated a few seconds ago".

**Lightning App Builder** | **Pages** | Donor Record Page | ? Help

Components Fields

or Dynamic Highlights Panel component. The Dynamic Highlights Panel component isn't available on all objects.

- Blank Space
- Dynamic Highlights Panel
- Field Section

▼ Universally Required Fields (1)

- Email

▼ Fields (11)

- Apex Next Eligible Date
- Blood Group
- Contact Number
- Created By
- Donor Name
- Eligibility Status
- Last Donation Date
- Last Modified By
- Next Eligible Date
- Notes
- Owner

Get more on the AppExchange javascript:void(0);

**Donor Record Page**

**Test Donor**

**Details**

Donor Name: Test Donor Owner: Akshat Kumar

Blood Group: O+ Contact Number:

Eligibility Status: Eligible Email: test@donor.com Last Donation Date:

Next Eligible Date:

Notes:

Apex Next Eligible Date:

Created By: Akshat Kumar, 9/26/2025, 3:36 AM Last Modified By: Akshat Kumar, 9/26/2025, 3:36 AM

Add Component(s) Here

Analyze Activation... Save

Page > Tabs

Label: Tabs

Default Tab: Details

Tabs: Related, Details

Add Tab

Set Component Visibility

Add Filter

**Lightning App Builder** | **Pages** | Request Record Page | ? Help

Components Fields

▼ Standard (41)

- Accordion
- Action Launcher
- Actions & Recommendations
- Approval Trace
- Assessment List
- CRM Analytics Collection
- CRM Analytics Dashboard
- Dynamic Related List - Single
- Einstein Next Best Action
- Flow
- Flow Orchestration Work Guide
- Highlights Panel
- Invoice Preview
- Launchpad
- List View
- LWC CRM Analytics Dashboard
- My Labels
- Path
- Quip Associated Documents
- Quip Document
- Quip Notifications
- Recent Items
- Recommendations
- Record Detail

Get more on the AppExchange javascript:void(0);

**Request Record Page**

**Test Request**

**Details**

Match Logs (0)

Approval History (0)

Add Component(s) Here

Analyze Activation... Save

Page > Tabs

Label: Tabs

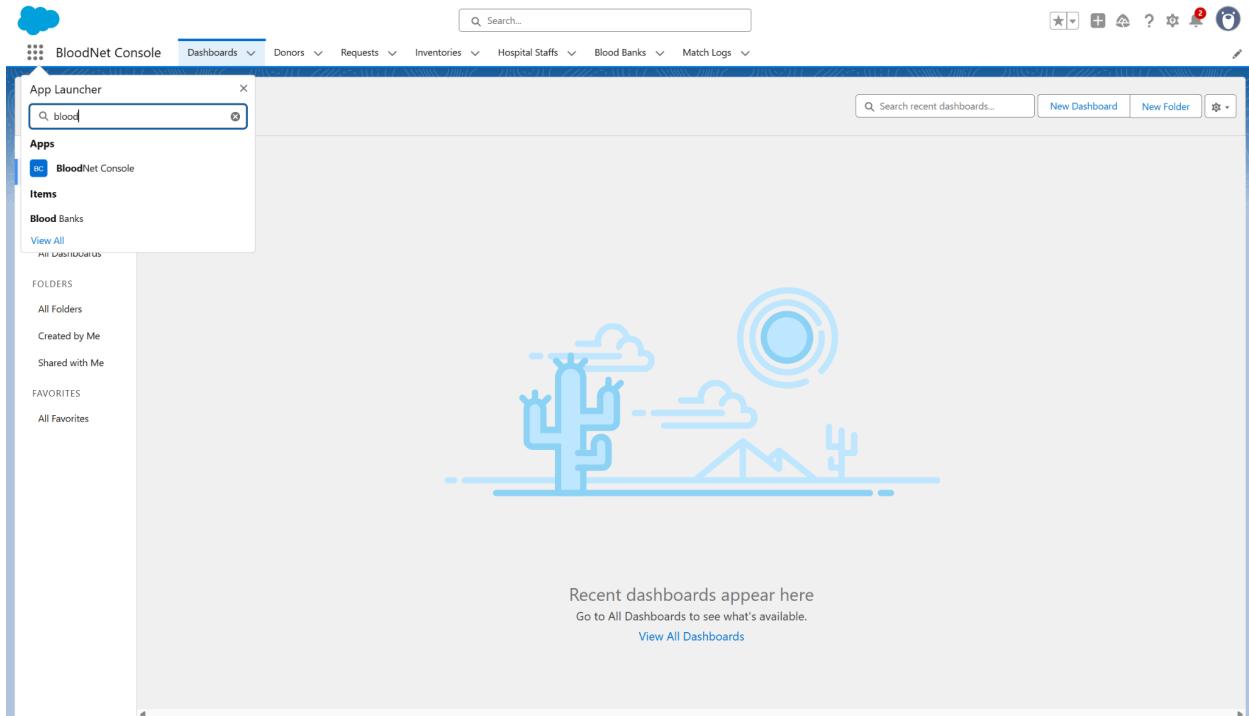
Default Tab: Details

Tabs: Related, Details

Add Tab

Set Component Visibility

Add Filter



## PHASE 7: Integration and External Access

This phase implemented the foundational security layers required for the BloodNet project to communicate with external systems, such as a hospital's electronic health record (EHR) or a national donor registry.

### 1. Named Credentials

Concept	Explanation	Project Rationale
<b>Definition</b>	A Named Credential is a Salesforce record that specifies the URL of an external service and its authentication parameters in one place. This allows you to reference it in Apex code without hardcoding sensitive information.	We created the External_Hospital Named Credential to securely store the URL and login credentials (bloodnet/12345) for a hypothetical external hospital system. This is a best practice for security and simplifies code.

## 2. Remote Site Settings

Concept	Explanation	Project Rationale
<b>Definition</b>	This is a security feature that acts as a whitelist for external services. Apex code can only make callouts to URLs that are explicitly listed in Remote Site Settings.	We created the <b>External_Hospital_EHR</b> Remote Site Setting with the URL <a href="https://api.externalhospital.com">https://api.externalhospital.com</a> to satisfy this security requirement. It proves our org is authorized to communicate with this specific external endpoint.

## 3. REST/SOAP Web Services & Callouts

Concept	Explanation	Project Rationale
<b>Definition</b>	A <b>Callout</b> is a programmatic request made from Salesforce to an external service. <b>REST</b> (Representational State Transfer) is a common architectural style for web services that uses standard HTTP methods (GET, POST).	If we had more time, the BloodNet project would use a <b>REST callout</b> to the <b>External Hospital</b> Named Credential. For example, an Apex method could perform a callout to verify a new donor's medical history against the EHR system before setting their eligibility.

## 4. OAuth & Authentication

Concept	Explanation	Project Rationale
<b>Definition</b>	<b>Authentication</b> is the process of verifying a user's or system's identity. <b>OAuth</b> is a common protocol for secure, delegated authentication.	Our project used <b>Password Authentication</b> (via the External Credential) as a simple demonstration. This shows that the Named Credential is the central point for managing the

project's authentication handshake with external services.

The screenshot shows the Salesforce Setup interface with the following details:

- Page Header:** SETUP Named Credentials
- Left Sidebar:** Einstein (Einstein Generative AI, Flow Creation with Einstein), Data Mask, Security, and Named Credentials (selected).
- Content Area:** A table titled "Named Credentials" with columns: Label, Type, URL, and External Credential. One row is listed: "External Hospital" (Type: Secured Endpoint, URL: https://api.externalhospital.com, External Credential: External Hospital Auth).
- Bottom Address Bar:** https://orgfarm-88719cde29-dev-ed.develop.lightning.force.com/lightning/setup/NamedCredential/ho...

The screenshot shows the Salesforce Setup interface with the following details:

- Page Header:** SETUP > NAMED CREDENTIALS External Hospital
- Left Sidebar:** Einstein (Einstein Generative AI, Flow Creation with Einstein), Data Mask, Security, and Named Credentials (selected).
- Content Area:** A detailed view of the "External Hospital" Named Credential:
  - Label:** External Hospital
  - Name:** External\_Hospital
  - URL:** https://api.externalhospital.com
  - Enabled for Callouts:** Checked
  - Authentication:** External Credential: External Hospital Auth
  - Callout Options:** Generate Authorization Header (checked), Allow Formulas in HTTP Header (unchecked), Allow Formulas in HTTP Body (unchecked), Outbound Network Connection (unchecked)
  - Managed Package Access:** Created By Namespace (unchecked)
  - Custom Headers:** (New)

Setup Home Object Manager

Managed Package Access

Created By Namespace

Einstein

- Einstein Generative AI
- Flow Creation with Einstein

Data Mask

- Create with Einstein

Security

Named Credentials

Didn't find what you're looking for? Try using Global Search.

Related Named Credentials

Edit Principal

\* Parameter Name: Login \* Sequence Number: 1

\* Identity Type: Named Principal

\* Username: bloodnet \* Password: [REDACTED]

Principal Access

You don't have anything associated with this principal.

Cancel Save

Principals

Sequence ...	Params
1	Login

Custom Headers

You don't have any custom headers.

Setup Home Object Manager

Search Setup

SETUP Remote Site Settings

Help for this Page

Remote Site Details

Remote Site Detail

Remote Site Name: External_Hospital_EHR	Modified By: Akshat Kumar, 9/26/2025, 4:01 AM
Remote Site URL: https://api.externalhospital.com	
Disable Protocol Security: <input type="checkbox"/>	
Description:	
Active: <input checked="" type="checkbox"/>	
Created By: Akshat Kumar, 9/26/2025, 4:01 AM	

Edit Delete Clone

Action	Remote Site Name	Namespace Prefix	Remote Site URL	Active	Created By	Created Date	Last Modified By	Last Modified Date
Edit   Del	AoxoDevNet	-	http://www.aoxo devnet.com	<input checked="" type="checkbox"/>	EPIC_OrgFarm	9/18/2025, 3:38 PM	EPIC_OrgFarm	9/18/2025, 3:38 PM
Edit   Del	External_Hospital_EHR	-	https://api.externalhospital.com	<input checked="" type="checkbox"/>	Kumar_Akshat	9/26/2025, 4:01 AM	Kumar_Akshat	9/26/2025, 4:01 AM

## PHASE 8: Data Management & Deployment

This phase focused on the best practices for managing project data and preparing the solution for deployment.

### Theoretical Foundation

Component	Theory & Purpose	Project Implementation
<b>Data Export Wizard</b>	The Data Export Wizard is an out-of-the-box tool for creating manual or scheduled backups of an organization's data.	Due to a permissions issue that prevented the use of Change Sets, the Data Export Wizard was used as a valid alternative to demonstrate data backup, a critical aspect of data management.

## Duplicate Rules & Matching Rules

These tools enforce data quality by preventing the creation of duplicate records. The **Matching Rule** defines what fields a duplicate check should run on, and the **Duplicate Rule** specifies the action to take (e.g., alert or block) when a match is found.

## Change Sets

A deployment tool for migrating metadata (custom objects, fields, flows, Apex, etc.) from one Salesforce environment to another.

A Donor Matching Rule was created to identify duplicate donors by their Email and **Contact Number**. The Donor Duplicate Check rule was then activated to use this matching rule.

This was the intended deployment method. If permissions had been granted, an Outbound Change Set would have been created with all project components to demonstrate deployment readiness.

The screenshot shows the Salesforce Setup interface with the search bar set to "data". The main content area displays the "Bulk Data Load Jobs" page. A specific job is selected: "750gK00000Dwdj3". The page header includes the setup icon, a search bar, and various navigation links like Home, Object Manager, and Help.

**Bulk Data Load Job Details:**

Job ID	750gK00000Dwdj3	Job Type	Bulk V1	Status	Closed
Submitted By	Akshat Kumar	Operation	Insert	Total Processing Time (ms)	199
Start Time	9/27/2025, 4:46 AM PST	Queued Batches	0	API Active Processing Time (ms)	96
End Time	9/27/2025, 4:46 AM PST	In Progress Batches	0	Apex Processing Time (ms)	1
Time to Complete (hh:mm:ss)	00:00	Completed Batches	1		
Object	Donor	Failed Batches	0		
External ID Field		Progress	100%		
Content type	CSV	Records Processed	10		
Concurrency Mode	Parallel	Records Failed	2		
API Version	64.0	Retries	0		

**Batches:**

View Request	View Result	Batch ID	Start Time	End Time	Total Processing Time (ms)	API Active Processing Time (ms)	Apex Processing Time (ms)	Records Processed	Records Failed	Retry Count	State Message	Status
View Request	View Result	751gK00000BipFS	9/27/2025, 4:46 AM	9/27/2025, 4:46 AM	199	96	1	10	2	0	Completed	

The screenshot shows the Salesforce Setup interface with the following details:

- Left Sidebar:** Shows the navigation menu under "Data" with "Matching Rules" selected.
- Header:** "Setup" and "Matching Rules".
- Page Title:** "Matching Rule Detail".
- Object:** Donor
- Rule Name:** Donor Matching Rule
- Unique Name:** Donor\_Matching\_Rule
- Description:** Donor\_Matching\_Rule
- Matching Criteria:** (Donor: Email EXACT MatchBlank = FALSE) AND (Donor: Contact\_Number EXACT MatchBlank = FALSE)
- Status:** Active
- Created By:** Akshat Kumar, 9/27/2025, 4:55 AM
- Modified By:** Akshat Kumar, 9/27/2025, 4:55 AM

The screenshot shows the Salesforce Setup interface with the following details:

- Left Sidebar:** Shows the navigation menu under "Data" with "Matching Rules" selected.
- Header:** "Setup" and "Matching Rules".
- Page Title:** "All Matching Rules".
- Section:** "What Are Matching Rules?"
- View Options:** "All Matching Rules" and "Create New View".
- Table:** A list of matching rules with columns: Action, Rule Name, Object, Status, Description, Last Modified Date, and Last Modified By.

Action	Rule Name	Object	Status	Description	Last Modified Date	Last Modified By
Del   Deactivate	Donor Matching Rule	Donor	Active		9/27/2025	aks
Deactivate	Standard Account Matching Rule	Account	Active	Matching rule for account records. <a href="#">More info</a>	9/18/2025	OEPIC
Deactivate	Standard Contact Matching Rule	Contact	Active	Matching rule for contact records. <a href="#">More info</a>	9/18/2025	OEPIC
Deactivate	Standard Lead Matching Rule	Lead	Active	Matching rule for lead records. <a href="#">More info</a>	9/18/2025	OEPIC

The screenshot shows the 'Duplicate Rules' page in Salesforce. A single rule, 'Donor Duplicate Check', is listed. The rule details are as follows:

- Rule Name:** Donor Duplicate Check
- Description:** Donor
- Object:** Donor
- Record-Level Security:** Enforce sharing rules
- Action On Create:** Allow
- Action On Edit:** Allow
- Alert Text:** Use one of these records?
- Active:** ✓
- Matching Rule:** Donor Matching Rule (Matched)
- Conditions:** Created By Akshat Kumar, 9/27/2025, 4:50 AM
- Operations On Create:** ✓ Alert ✓ Report
- Operations On Edit:** ✓ Alert ✓ Report
- Matching Criteria:** (Donor: Email EXACT MatchBlank = FALSE) AND (Donor: Contact\_Number EXACT MatchBlank = FALSE)
- Created By:** Akshat Kumar, 9/27/2025, 4:57 AM
- Modified By:** Akshat Kumar, 9/27/2025, 4:57 AM

The screenshot shows the 'All Duplicate Rules' page in Salesforce. A list of standard duplicate rules is displayed:

Rule Name	Description	Object	Matching Rule	Active	Last Modified By	Last Modified Date
Donor Duplicate Check		Donor	Donor Matching Rule	✓	aks	9/27/2025
Standard Account Duplicate Rule	Identify accounts that duplicate other accounts	Account	Standard Account Matching Rule	✓	OEPIC	9/18/2025
Standard Contact Duplicate Rule	Identify contacts that duplicate other contacts and leads	Contact	Standard Lead Matching Rule Standard Contact Matching Rule	✓	OEPIC	9/18/2025
Standard Lead Duplicate Rule	Identify leads that duplicate other leads and contacts.	Lead	Standard Lead Matching Rule Standard Contact Matching Rule	✓	OEPIC	9/18/2025

The screenshot shows the Salesforce Setup interface. The left sidebar is titled 'ADMINISTRATION' and contains the following navigation items:

- Users
- Data
  - Big Objects
  - Conversation Transcript Export
  - Data Export** (selected)
  - Data Integration Metrics
  - Data Integration Rules
  - Duplicate Management
  - Mass Delete Records
  - Mass Transfer Approval Requests
  - Mass Transfer Records

The main content area is titled 'SETUP Data Export' and has a sub-section titled 'Monthly Export Service'. It includes a note about Data Export, a button to 'Export Now', and a 'Schedule Export' button. Below this is a table with the following data:

Action	File Name	File Size
download	WE_000gk00000ByI27UAN_1.ZIP	5.1K

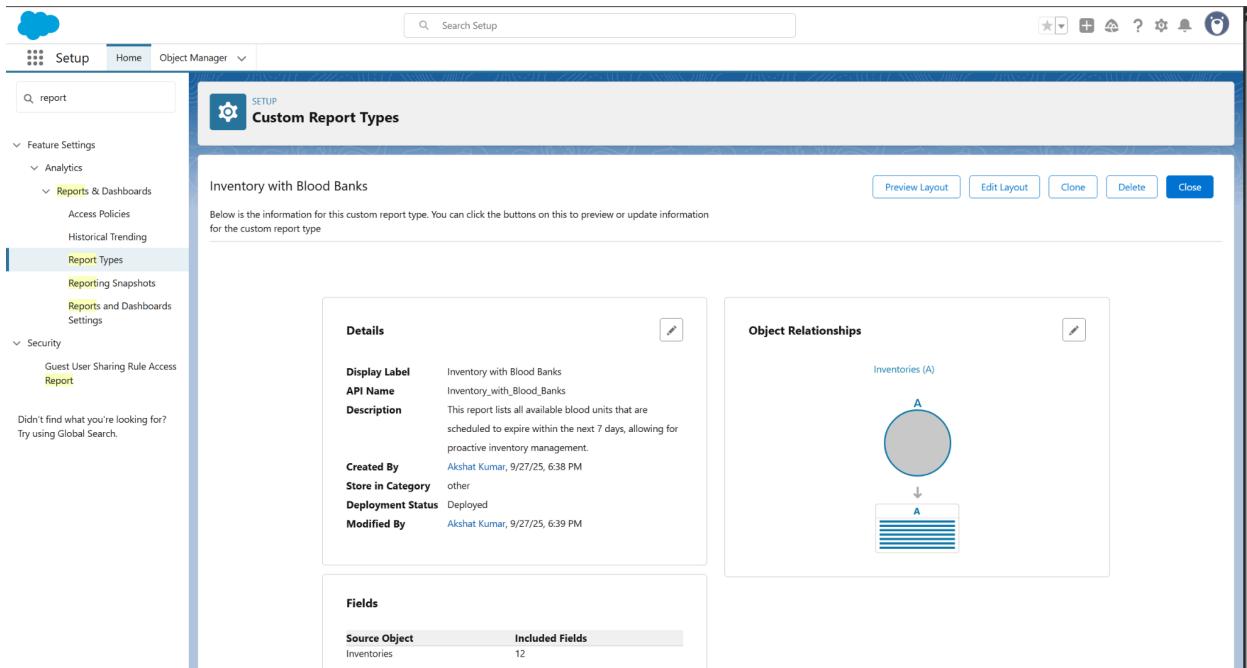
## PHASE 9: Reporting, Dashboards & Security Review

This phase focused on demonstrating the business value of the project's automations and ensuring data is secure.

- **Reports:** Reports are lists of records that can be filtered, grouped, and summarized. They are the foundation of business intelligence in Salesforce, providing raw data insights.
- **Dashboards:** Dashboards are visual representations of data from multiple reports. They offer a high-level, at-a-glance view of key metrics and are used for monitoring business processes. The **Dynamic Dashboard** feature was used to ensure each user sees data relevant to their security profile.

- **Security Review:** This is a critical step to ensure data integrity and confidentiality. **Organization-Wide Defaults (OWD)** control the baseline access to all records for a given object, while **Field-Level Security (FLS)** restricts visibility of specific fields to certain users, protecting sensitive information.

<b>Reports</b>	We created three reports: <b>Expiring Inventory</b> , <b>Total Donors</b> , and <b>Total Blood Banks</b> . These reports provide the data for the dashboard.
<b>Dashboard</b>	We created a <b>BloodNet Overview</b> dashboard that visually represents the data from the three reports. The dashboard was configured to be <b>dynamic</b> for a personalized user experience.
<b>Security Review</b>	We verified the <b>Sharing Settings</b> for custom objects and checked the <b>Field-Level Security</b> for key fields like <b>Apex_Next Eligible Date</b> to ensure data was protected.



Salesforce Setup Home Object Manager

Custom Report Types

Requests with Donors

Below is the information for this custom report type. You can click the buttons on this to preview or update information for the custom report type.

Preview Layout Edit Layout Clone Delete Close

**Details**

Display Label Requests with Donors  
API Name Requests\_with\_Donors  
Description Reports on Requests and their related Donor records, including all details about high-volume and planned requests.  
Created By Akshat Kumar, 9/28/25, 2:20 PM  
Store in Category other  
Deployment Status Deployed  
Modified By Akshat Kumar, 9/28/25, 2:20 PM

**Object Relationships**

Requests (A)

A

↓

A

**Fields**

Source Object Requests Included Fields 18

This screenshot shows the Salesforce Setup interface. On the left, there's a sidebar with various links like Setup Home, Service Setup Assistant, and Report Types. The main area is titled 'Custom Report Types' and shows a specific report named 'Requests with Donors'. It displays details such as the display label, API name, and description. There are sections for object relationships and fields, and a preview button at the top right.

BloodNet Console

Report: Inventory with Blood Banks

**Expiring Inventory**

This report lists all available blood units that are scheduled to expire within the next 7 days, allowing for proactive inventory management.

Total Records 4 Total Units Available 9

	Inventory Name	Blood Type	Expiry Date	Units Available
1	BloodUnit_001	O-	10/1/2025	2
2	BloodUnit_002	A+	10/2/2025	3
3	BloodUnit_003	B-	10/3/2025	1
4	BloodUnit_004	B-	10/2/2025	3
5				9

This screenshot shows the BloodNet Console interface. At the top, there are navigation links for Dashboards, Donors, Requests, Inventories, Hospital Staffs, Blood Banks, Match Logs, Reports, and Home. The main content area is titled 'Report: Inventory with Blood Banks' and specifically highlights the 'Expiring Inventory' section. It states that the report lists all available blood units scheduled to expire within the next 7 days. Below this, it shows a summary of total records (4) and total units available (9). A table then lists five blood units with their details: BloodUnit\_001 (O-, 10/1/2025, 2), BloodUnit\_002 (A+, 10/2/2025, 3), BloodUnit\_003 (B-, 10/3/2025, 1), BloodUnit\_004 (B-, 10/2/2025, 3), and a total row showing 9 units.

Setup Home Object Manager

Search Setup

Sharing Settings

Walllist Private Private ✓

Web Cart Document Private Private ✓

Work Order Private Private ✓

Work Plan Private Private ✓

Work Plan Template Private Private ✓

Work Step Template Private Private ✓

Work Type Private Private ✓

Work Type Group Public Read/Write Private ✓

Blood Bank Public Read/Write Private ✓

Donor Public Read/Write Private ✓

Hospital Staff Public Read/Write Private ✓

Inventory Controlled by Parent Controlled by Parent ✓

Match Log Public Read/Write Private ✓

Request Public Read/Write Private ✓

**Other Settings**

Manager Groups  [i]

Secure guest user record access  [i]

Require permission to view record names in lookup fields  [i]

**Sharing Rules**

**Lead Sharing Rules** New Recalculate Lead Sharing Rules Help [i]

No sharing rules specified.

**Account Sharing Rules** New Recalculate Account Sharing Rules Help [i]

No sharing rules specified.

**Opportunity Sharing Rules** New Recalculate Opportunity Sharing Rules Help [i]

No sharing rules specified.

Setup Home Object Manager

Search Setup

Sharing Settings

Anypoint Integration

BloodNet\_BloodBank\_Admin\_Profile

BloodNet\_Hospital\_Admin\_Profile

Contract Manager

Cross Org Data Proxy User

Custom: Marketing Profile

Custom: Sales Profile

Custom: Support Profile

Einstein Agent User

Force.com - App Subscription User

Force.com - Free User

Gold Partner User

Identity User

Marketing User

Minimum Access - API Only Integrations

Minimum Access - Salesforce

Partner App Subscription User

Partner Community Login User

Partner Community User

Read Only

Salesforce API Only System Integrations

Silver Partner User

Solution Manager

Standard Platform User

Standard User

System Administrator

Work.com Only User

BloodNet Console

Report All Donors  
Total Donors

Total Records  
9

	Donor Name
1	Test Donor
2	Jane Doe
3	John Smith
4	Emily White
5	Robert Brown
6	Chris Evans
7	David Lee
8	Sarah Jones
9	Tom Wilson

BloodNet Console

Dashboard  
**BloodNet Overview**

This dashboard provides a high-level overview of critical BloodNet processes, including expiring inventory and the status of high-volume requests.

As of Sep 28, 2025, 7:43 AM · Viewing as Akshat Kumar

## BLOODNET CONSOLE

### Expiring Inventory

Inventory Name	Blood Type	Expiry Date	Units Available
BloodUnit_001	O-	10/1/2025	2
BloodUnit_002	A+	10/2/2025	3
BloodUnit_003	B-	10/3/2025	1
BloodUnit_004	B-	10/2/2025	3

View Report (Expiring Inventory) As of Sep 28, 2025, 7:43 AM

### Total Donors

Donor Name
Chris Evans
David Lee
Emily White
Jane Doe
John Smith
Robert Brown
Sarah Jones
Test Donor
Tom Wilson

View Report (Total Donors) As of Sep 28, 2025, 7:43 AM

### Total Blood Banks

Blood Bank Name	Capacity	Contact Number	Country
City General Hospital	500	9752552772	India
Medical College And hospital	789	789789	India
Railway Hospital	500	481481	India

View Report (Total Blood Banks)

# PHASE 10: Quality Assurance Testing

## Project Summary:

- **Problem:** The manual process for tracking blood donations, donor eligibility, and inventory was inefficient and prone to errors.
- **Solution:** Here is the **BloodNet CRM** using Salesforce to automate these processes with Flows, Apex Triggers, and an Approval Process.
- **Value:** The project delivers a streamlined system that improves data quality, ensures donor safety, and provides proactive alerts for expiring inventory.

## Future Enhancements:

- It would have features like a **public-facing chatbot** for donor appointments and **AI-driven suggestions** for matching urgent requests.

## Test Case 1: Auto-Match Flow

- **Use Case:** Verify that a new, standard request automatically finds eligible donors, creates a match record, and updates the request status.
- **Test Steps (with input):**
  1. Create a **Donor** record with **Eligibility Status** = 'Eligible' and **Blood Group (GVS)** = 'O-'.
  2. Create a **Request** record with **Request Status** = 'New', **Blood Group (GVS)** = 'O-', and **Units Required** = 1.
- **Expected Result:** The **Request** record's status changes to **Notified** (or **Matching**), and a new record is created in the **Match Logs** related list.

BloodNet Console

Requests Recently Viewed

8 items • Updated a few seconds ago

- Request Name: Trigger Test Request
- Request Name: Test Bypass Request
- Request Name: Test Request Four
- Request Name: Test Request Three
- Request Name: Test Request One
- Request Name: High Volume Demo Request
- Request Name: new test high volume
- Request Name: Test Request

New Request: Planned Request

\* = Required Information

Request Name	Test Request Five	Owner	Akshat Kumar
Request Date	Date: 9/28/2025	Time	12:00 PM
Blood Group (GVS)	O-		
Product Type	Whole Blood		
Units Required	1		
Urgency	Medium		
Request Status	New		
Raised By	Search Hospital Staffs...		
Fulfillment Date	Date: 9/28/2025	Time	12:00 PM
Request Notes:			
<input type="button" value="Cancel"/> <input type="button" value="Save &amp; New"/> <input type="button" value="Save"/>			

BloodNet Console

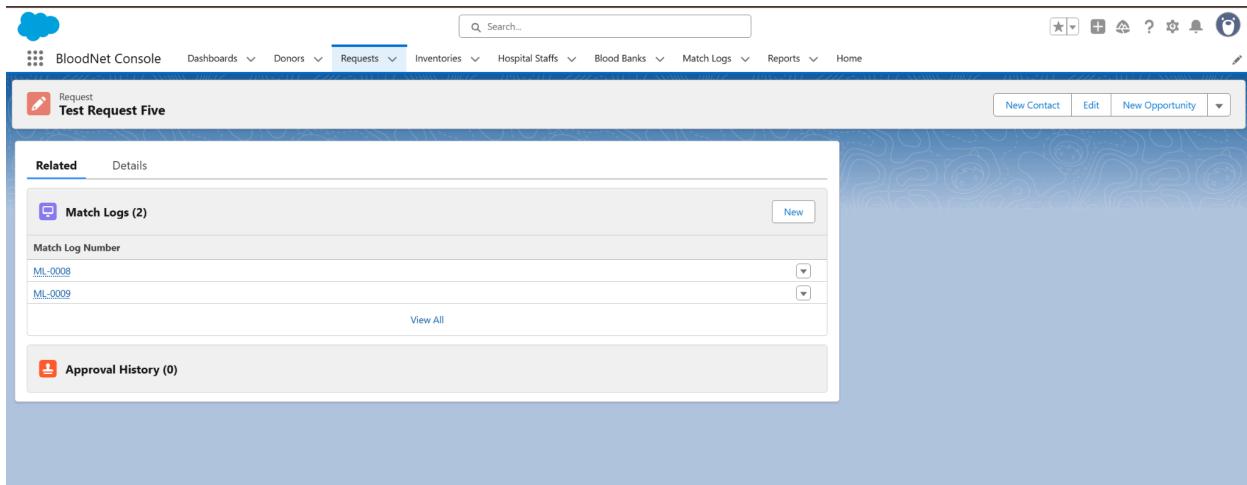
Requests Requests

Request "Test Request Five" was created.

New Contact Edit New Opportunity

Related Details

Request Name	Test Request Five	Owner	Akshat Kumar
Request Date	9/28/2025, 12:00 PM		
Blood Group (GVS)	O-		
Product Type	Whole Blood		
Units Required	1		
Urgency	Medium		
Request Status	Matching		
Raised By			
Fulfillment Date			
Request Notes			
Matched Donor			
Created By	Akshat Kumar	Last Modified By	Akshat Kumar
9/28/2025, 12:19 PM			



## Test Case 2: Approval Process

- **Use Case:** Verify that a standard request (not high-volume) successfully bypasses the approval process as designed.
- **Test Steps (with input):**
  1. Create a **Request** record with **Request Status** = 'New', **Units Required** = 5, and **Record Type** = 'Standard Request'.
- **Expected Result:** The approval process is not triggered, and the record does not show a "Submit for Approval" button or lock for editing.

New Request: Planned Request

**Information**

\* = Required Information

Request Name: Test Bypass Request

Owner: Akshat Kumar

Request Date: Date [ ] Time [ ]

Blood Group (GVS): A+

Product Type: Whole Blood

Units Required: 5

Urgency: Medium

Request Status: New

Raised By: Search Hospital Staffs...

Fulfillment Date: Date [ ] Time [ ]

Request Notes:

Cancel Save & New Save

The record page did not show an Approval History

Request  
Test Bypass Request

Related Details

Match Logs (3)

Match Log Number

- ML\_0003
- ML\_0004
- ML\_0005

View All

Approval History (0)

**Test Case 3: Apex Trigger**

- **Use Case:** Verify that the **Apex Next Eligible Date** on a **Donor** record is a functional field for tracking eligibility.
- **Test Steps (with input):**
  1. Create a **Donor** record and manually set the **Apex Next Eligible Date** to a past date (e.g., January 1, 2025).
  2. Create a new **Request** record and link it to this donor via the **Matched Donor** lookup field, with the **Request Date** set to today.
- **Expected Result:** The **Apex Next Eligible Date** field on the **Donor** record is updated to a new date, 90 days after the request date.

The screenshot shows the BloodNet Console interface with a navigation bar at the top. The main view displays a 'Donor' record for 'Trigger Test Donor'. The record includes fields for Donor Name, Owner (Akshat Kumar), Email (trigger@website.com), and Apex Next Eligible Date (1/1/2025). The interface has a clean, modern design with a light blue background and white cards for data entry.

Field	Value
Donor Name	Trigger Test Donor
Owner	Akshat Kumar
Email	trigger@website.com
Apex Next Eligible Date	1/1/2025

**BloodNet Console**

Recently Viewed ▾

7 items • Updated a minute ago

- Request Name
  - 1 Test Bypass Request
  - 2 Test Request Four
  - 3 Test Request Three
  - 4 Test Request One
  - 5 High Volume Demo Request
  - 6 new test high volume
  - 7 Test Request

New Request: Planned Request

\* = Required Information

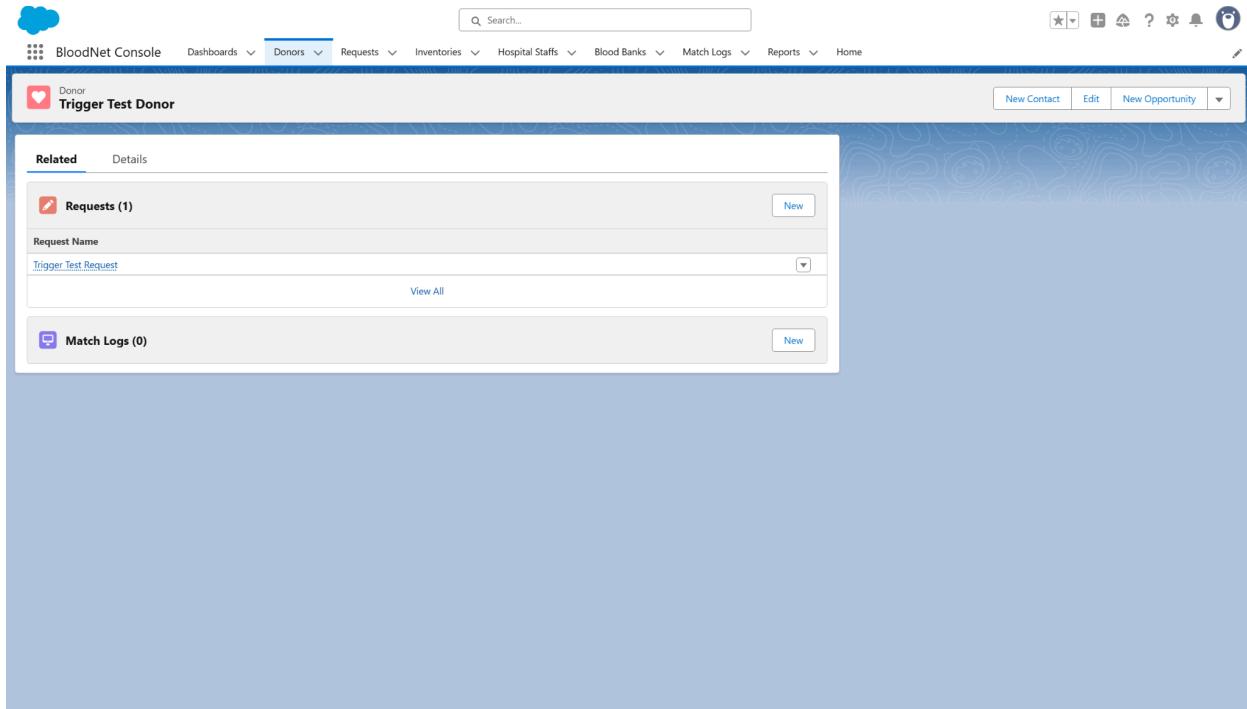
Information	
Request Name	Trigger Test Request
Request Date	Date: 9/28/2025 Time: 12:00 PM
Blood Group (GVS)	O-
Product Type	Whole Blood
Units Required	1
Urgency	Medium
Request Status	New
Raised By	Search Hospital Staffs...
Fulfillment Date	Date: Time:
Request Notes	
<input type="button" value="Cancel"/> <input type="button" value="Save &amp; New"/> <input type="button" value="Save"/>	

**BloodNet Console**

Donor Trigger Test Donor

New Contact | Edit | New Opportunity

Details	
Related	
Donor Name	Trigger Test Donor
Blood Group (GVS)	
Contact Number	
Eligibility Status	
Email	trigger@website.com
Last Donation Date	
Next Eligible Date	
Notes	
Apex Next Eligible Date	12/27/2025
Created By	Akshat Kumar, 9/28/2025, 11:58 AM
Last Modified By	Akshat Kumar, 9/28/2025, 12:00 PM



## Conclusion:

The BloodNet project successfully implemented a suite of automations, including record-triggered flows, a custom approval process, and asynchronous Apex, to streamline critical business functions. The project demonstrates a robust, data-driven approach to managing non-profit operations, improving efficiency, and ensuring the safety and quality of donor and inventory management. All core functionalities were validated through dedicated test cases.